

Quarterly Newsletter  
CHAIR  
**iDANAE**  
INTELLIGENCE · DATA · ANALYSIS · STRATEGY

4Q25

LLM specialisation: Towards task- and domain-specific optimisation



POLITÉCNICA

UNIVERSIDAD  
POLITÉCNICA  
DE MADRID

**MSO** Management  
Solutions  
Making things happen



# Introduction

As the use of artificial intelligence (AI) moves from the experimental phase to actual deployment, a clear pattern is emerging: **general-purpose language models (LLMs)** are not entirely sufficient for tasks that require a deep understanding of a domain. Many applications need models capable of interpreting specialised terminology or reasoning with information that is not easily accessible by general-purpose models. This is particularly relevant in technical and specialised fields, such as **sustainability (ESG), credit risk, legal analysis, and healthcare**, among others, where accuracy and context are critical.

This realisation has driven growing interest in **specialised LLMs**: models designed to excel within a defined area of knowledge. The current wave of innovation reflects a broader effort to **balance general linguistic ability with domain-specific accuracy**.

There are three **main technical approaches** commonly used to make expert knowledge available and incorporate it into LLMs:

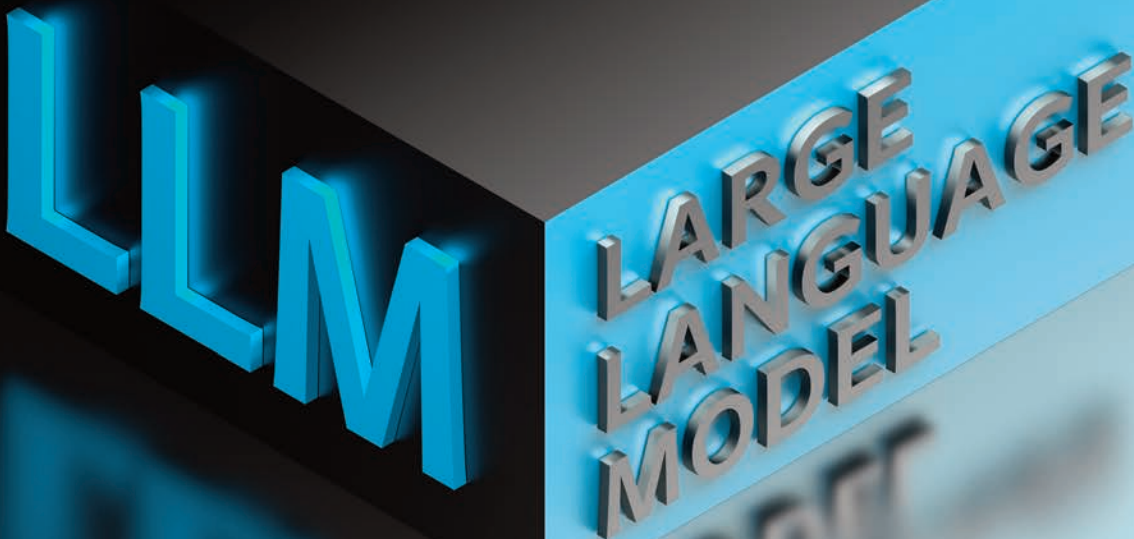
1. **Scaling general models**, leveraging the broad knowledge embedded in large LLMs, which may partially cover certain specialised domains.
2. **Fine-tuning**, which consists of adapting a pre-trained model by modifying its weights—either totally or partially—with domain-specific data, or by using already fine-tuned alternatives.
3. **Knowledge augmentation**, using methods such as **RAG (Retrieval-Augmented Generation)**, which enriches responses by retrieving relevant external information, and **ICL (In-Context Learning)**, which allows the model to infer patterns or styles directly from the prompt content.

In practice, these approaches are often combined, as each offers different advantages in terms of adaptability, accuracy, and efficiency.

In addition, there is the possibility of using integrated tools and frameworks, which offer ready-made solutions for creating specialised agents or workflows without the need to completely retrain a model.

Finally, it is worth highlighting some trends in the field of domain expert models, both in the academic and business fields. Relevant examples include the growing use of SLMs (Small Language Models) and Model Routers.

This publication explores how specialised LLMs are transforming the landscape of AI applications, enabling more reliable and contextualised results in complex and knowledge-intensive domains.



# Technical Review

## Scaling general models

As previously noted, there are multiple strategies for adapting an LLM to a specific field of knowledge or task. Traditionally, the dominant paradigm has been "**bigger is better**" [1] [2] based on the premise that larger models offer **greater generalisation** and versatility. Indeed, so-called generalist **frontier models** tend to **perform outstandingly well on a wide variety of tasks**, which in many cases **allows them to be used directly**, without the need for additional customisation.

However, even among these enormous models, each one stands out in different areas[3], and there is still much **room for improvement**, especially in **highly specialised sectors** [4], characterised by a scarcity of public data or the complex interpretation of such data. Some examples of sectors with these characteristics are law, medicine, and finance.

Furthermore, large-scale models often involve **high costs**. In many production scenarios, this approach may not be optimal, especially when tasks are repetitive or limited in scope. In contrast, **there are smaller models** that, when

deployed in a controlled manner or through hybrid solutions, **offer comparable** or even better **performance** [5] **at a fraction of the cost** [6]. Thus, although large models can successfully solve a wide range of tasks, their use is not always the most appropriate option from the perspective of resource optimisation, scalability and, in some cases, even accuracy.

Another compelling reason why model specialisation is recommended in many cases is the **risk of hallucination** [7], particularly in those domains where LLMs have limited knowledge or show low confidence in their predictions. This phenomenon, derived from the probabilistic nature of the model, can lead to the generation of **incorrect** or invented **information** that appears to be true. In environments where accuracy is critical, this risk can become unacceptable[8].

That is why model **fine-tuning** [9], **ICL** [10], and **RAG** [11] have gained much relevance, as they make models **more efficient and accurate**. Below, we will present the fundamentals of each of these approaches, followed by a comparison that will allow us to understand in which situations it is most appropriate to apply each one.



## Fine-tuning

Fine-tuning a language model consists of **training a pre-trained model** (such as GPT-5, Claude Sonnet 4.5, etc.) using proprietary data to adapt it to a specific use case.

One might wonder whether it would be possible to train an **expert model from scratch**, rather than performing fine-tuning. While it is true that it is during pre-training that an LLM acquires most of its knowledge [12], this stage is **difficult to replicate** in most practical scenarios, as it requires an enormous **amount of energy, data and infrastructure** [13]. As a reference, it is estimated that the training of large-scale models such as GPT-4 exceeded **£100 million**, according to statements by Sam Altman (CEO of OpenAI). Even considerably smaller models require very high investments, with computing costs ranging from \$100,000 to \$500,000, not including the costs associated with data development, cleansing and storage [14].

Performing a **complete fine-tuning**, i.e., modifying all the weights of a model, is also **very costly**. According to [15], completely fine-tuning a model on the scale of Llama-3 would require storage systems with several petabytes of capacity, ultra-high-speed memory interconnections, and hundreds or even thousands of state-of-the-art GPUs. This type of operation is not only **economically unfeasible** for most organisations, but also **technically complex**, as it requires precise orchestration of communication and synchronisation between GPUs, as well as advanced training pipeline management to avoid bottlenecks and efficiency losses.

Furthermore, when performing a complete fine tuning, there risk of **catastrophic forgetting** [16] [17] is introduced, a phenomenon in which the model loses some of the previously acquired knowledge by over-specialising in a new task, thus compromising its versatility and generalisation capacity.

That is why, in the field of fine tuning, the concept of **PEFT (Parameter Efficient Fine Tuning)** has emerged [18]. The basic idea behind PEFT is to keep most of the parameters of the base model frozen, retraining only a small subset of new parameters that adjust its behaviour to the desired task or domain. Some of the most popular examples are **LoRA** (Low-Rank Adaptation) [19], **QLoRA** [20], and so-called **adapters** [21] [22].

## Examples of PEFT

### LoRA (Low-Rank Adaptation)

Instead of updating all the parameters of an LLM, LoRA injects small low-rank matrices into the model layers, which act as additional adjustments without modifying the original weights. IBM describes it as follows: "LoRA is a technique that adapts a large model by adding lightweight components to the original, rather than changing the entire model." This approach dramatically reduces training costs and memory consumption, achieving a significant increase in efficiency without sacrificing model performance [23].

### QLoRA

This is a variant of LoRA that combines its low-rank approach with quantisation techniques, reducing the numerical precision of the parameters to decrease memory usage without substantially compromising performance. Google Cloud recommends LoRA for speed and cost, and also notes that QLoRA consumes ~75% less GPU memory [24].

### Adapters

Adapters are small additional modules inserted into the transformer layers, designed to adjust the behaviour of the model without modifying its original weights. This approach allows switching between tasks or domains by simply replacing the corresponding modules. Although the addition of adapters introduces some additional latency during inference and leads to greater resource usage in training, it can offer superior performance compared to lighter techniques, as a higher percentage of parameters are trained [25].

## Context Engineering (ICL y RAG)

ICL and RAG are strategies encompassed within the concept of **context engineering** [26] [27], a discipline that seeks to optimise the amount and manner in which information is introduced into an LLM call to obtain the **most accurate and efficient response possible**. Context engineering allows the potential of an LLM to be fully exploited, enabling them to perform tasks that would be impossible or much less efficient without this approach [28] [29].

In recent years, this field has undergone **rapid evolution**, resulting in a proliferation of **specialised research disciplines**, including:

- ▶ **Context Retrieval and Generation:** this is the most relevant area for the purpose of this document. It focuses on determining which elements should be selected and presented to the model, prioritising those that are most relevant or informative [30]. This can be geared towards both incorporating external knowledge and inducing specific behaviours [31].
- ▶ **Context Processing:** Studies how to structure and integrate information so that the model processes it more efficiently, facilitating more accurate decisions. Techniques such as GraphRAG make use of advanced Context Processing [32].

- ▶ **Context Management:** Focuses on optimising the space available in the prompt through strategies such as text compression, context vectorisation or dynamic selection of fragments, in order to balance relevance and memory capacity [33].

The application of these concepts allows us to extract the maximum possible performance from the models with strategies such as the following:

- ▶ **In Context Learning (primarily Prompt Engineering):** ICL is based on the ability of LLMs to learn patterns, styles, or behaviours simply from the content of the prompt. On a practical level, ICL has evolved from simple chains of examples (few-shot prompting), in which the LLM is shown how it should behave, to techniques such as prompt chaining, dynamic prompting or I2CL (Iterative In Context Learning) [34] [35].
- ▶ **RAG:** This technique introduces an intermediate layer between user input and model generation, responsible for retrieving relevant information from an external knowledge base and providing it as context to the LLM. In this way, the model can reason about up-to-date and specific information. More advanced versions, such as GraphRAG [32] or HybridRAG [36], improve the process through Context Processing techniques to provide more coherent and verifiable responses.



With the arrival of the **Model Context Protocol (MCP)** [37], the ability of LLMs to invoke external tools, such as APIs, databases, calculators, or browsers, has been **standardised and unified**. This advance allows models to reason more effectively about information that exceeds their internal knowledge and lays the foundation for **Intelligent Agent Systems**, considered "the pinnacle of context learning" [26].

According to Douwe Kiela, one of the researchers who introduced the RAG technique and CEO of Contextual AI, MCP complements techniques such as RAG by providing a **cleaner and more efficient framework** for communication between databases and linguistic models [38]. In addition, it enables the use of **Multi-Agent Systems (MAS) for information retrieval**, which can improve the accuracy of RAG using a similar, but agentic, pipeline.

### **Comparison between the different options**

Once we have reviewed the main techniques that enable a model to specialise in a specific domain (ICL, RAG and fine tuning), we can proceed to compare them, given that each approach has **advantages and limitations** depending on the use case.

**An illustrative analogy** when comparing these techniques is that of several students with different study strategies facing an exam in which they will be asked about a book [39] [40].

The first student has read and studied the book, so they understand the concepts and the relationships between them: this represents **fine-tuning**. The second student has not studied, but during the exam they can consult the book, searching directly for the answers they need. This would be the equivalent of **RAG**. Finally, a third student has not read the book, but has received precise instructions from the teacher on how (though not what) to answer the questions; this case represents **ICL**.

**Depending on the complexity and objective** of the exam, each student will perform better or worse. For example, in a history exam with specific dates, RAG might work better, while in a mathematical reasoning exam, fine-tuning would be more appropriate. In a simple exam where the form of the answer matters, with more or less general knowledge, ICL would suffice and no additional effort would be required. **Each technique requires a different type of infrastructure and expenditure.**

### **Technical requirements for each strategy**

**ICL** stands out for its **simplicity, low implementation cost, and flexibility**, far surpassing the other two techniques in these aspects [41]. It does not require additional infrastructure, embedding generation, or training processes, making it a particularly attractive option for **simple cases, experimentation phases, or rapid adaptation to new contexts**.

However, **ICL performance tends to decrease as task complexity increases**, and depends heavily on both the quality of the prompt and the model's ability to handle long contexts. ICL is often considered the **first strategy to evaluate** before resorting to more costly techniques, as it allows hypotheses to be validated and useful results to be obtained with minimal effort [34].

When using **RAG, the technical and operational requirements are more demanding** [28]. It is necessary to have a high-quality **document base**, which will serve as a source of knowledge for the model. For the system to function properly, a **vector database** must be integrated to store the numerical representations of the texts, along with an ingestion and processing pipeline capable of dividing the documents into manageable fragments (chunks).

In addition, an **embedding model** is required to convert the text into semantic vectors, and a retrieval mechanism that, when faced with a query, identifies the most relevant fragments within the knowledge base. This process **inevitably introduces some latency**, as searching and retrieving information requires additional time before the model can generate a response.

A basic RAG environment can be implemented using managed services[42] (embeddings API and vector base in the cloud), so the **initial configuration does not have to be complex**, although it is **more expensive than ICL**. The main costs come from data storage, embedding generation, and the increase in the number of tokens processed in each query.

**Fine-tuning is the most demanding technique of the three, both in terms of time and resources** [43]. It requires a **large amount of high-quality data** that is well labelled and representative of the specific domain or task to be improved. Preparing this dataset and ensuring that it is **well structured, clean, and balanced** often requires a significant amount of time and has a direct impact on the quality of the final result.

In addition, the process requires an **adequate training infrastructure**, generally one or more GPUs, depending on the size of the base model [20]. The process also requires **specialised technical personnel** with experience in hyperparameter tuning, validation and model monitoring after deployment. These tasks are critical to avoid overfitting and ensure stable performance in production [44].

**Some frontier models** offer the possibility of **fine tuning from the API itself**, which solves many of the problems mentioned so far, but the cost of inference can increase [45]. Furthermore, this functionality is **not common**, and, for example, GPT-5 has not maintained this option, having focused its development on Agentic-AI-oriented functionalities.

## Performance of each technique according to use case

The ability of fine-tuning to **add new knowledge** is a matter of debate in the current literature, with some studies claiming that using this technique to teach factual or external information to the model may, in fact, increase the likelihood of generating **hallucinations** [46].

In general terms, if the objective is for an LLM to be able to use and reason about **new or updated data**, the literature indicates that RAG is the most appropriate option [47], even in highly specialised contexts [48].

Recent studies seem to agree that fine-tuning is more appropriate for teaching the model **"skills,"** while RAG is more powerful when it comes to incorporating **new data into the model** [49]. In addition, RAG has the added advantage of being able to **dynamically change** the knowledge that the LLM has access to, making it particularly useful in environments where knowledge changes frequently. In practice, although RAG alone tends to be the business standard [43], being sufficient for most use cases, **the combination of both techniques can offer better results** [39].

For example, in the legal field [43], it is useful to apply fine-tuning so that a model learns to communicate like a lawyer, adopting their terminology, argumentative style, and understanding of specific linguistic nuances. In other words, the model develops a 'legal way of thinking'. These types of skills cannot be achieved with RAG, as **RAG does not teach the model to reason** or internalise patterns of thought, but simply gives it access to information.

On the other hand, laws and regulations themselves should not be incorporated through fine-tuning, as this would require **massive amounts of data and eliminate the traceability of responses**. This knowledge would be introduced with RAG, which also offers the advantage of being able to easily update the system when laws change or become obsolete, without the need to retrain the model.

Thus, once it has been determined that ICL is not a viable option for the use case, **if the goal is for the LLM to acquire a skill**, specific behaviour, or a deeper understanding of certain concepts, the most appropriate strategy will be **PEFT**. Conversely, when the goal is for the model **to incorporate or use new data**, it will be more efficient and scalable to use **RAG** [43].

If there is not enough data available to perform PEFT without the risk of overfitting, but ICL does not achieve the necessary depth for the task, there are **intermediate techniques** that can serve as a solution, achieving a level of adaptation comparable to that of other fine-tuning variants and surpassing the accuracy and consistency of ICL [50] [51].

## Examples of integrated tools and frameworks

In addition to the approaches described above, it is important to consider the use of integrated **"as a service"** platforms and frameworks, which allow both the deployment of specialised models and the creation of expert agents within specific business or technical environments. These solutions facilitate the incorporation of advanced reasoning capabilities, access to corporate data, and regulatory compliance without requiring proprietary training infrastructure. Below are some representative examples:

- ▶ **Microsoft 365 Copilot – Agents and extensibility.** Allows the creation of specialised agents within the Microsoft ecosystem, which act on business data in Microsoft Graph, Word, Excel, Teams or external connectors. It supports two approaches: declarative agents (configuration of instructions, data, actions) and custom engine agents (external hosting, proprietary or specialised models) [52]. Some of the advantages include reduced complexity, increased speed to market, improved reliability and compliance, etc. Some examples that can be created are finance agents, invoice management, internal process automation, etc. [53].
- ▶ **Expert.ai – Vertical AI solution for specific domains.** Offers greater control, explainability and semantic accuracy for regulated or high-risk sectors (legal, compliance) [54].
- ▶ **Services:** There are some platforms such as Zfort Group [55] or Scopic [56] that offer customised or integrated developments for sectors such as healthcare, finance, legal, etc.
- ▶ **Creation of multimodal agents for complex domains.** These types of solutions illustrate where the market is heading: "domain expert" agents that combine base models + context + specialised tools.

### **Other trends: Model Routers and SLMs**

In the field of **cost and resource optimisation** when performing tasks with **Large Language Models**, the emergence of **model routers** stands out. These are a category of tools and architectures designed to **dynamically assign** requests to the most appropriate model according to the type of task, the complexity of the input and the cost per token. Most of these solutions can be understood as an extrapolation of the **Mixture of Experts (MoE)** paradigm[60] to the field of LLMs. In this approach, a **set of expert models**, each optimised for **specific** tasks or domains **and of small size**, generates predictions that are then weighted by a **routing model** (gating network model), which determines how much weight each expert should have in the final response.

Another notable trend is the **growing interest in SLMs** (Small Language Models) within the AI ecosystem [64]. SLMs are **lighter models** than large generalist LLMs. Unlike LLMs, SLMs are designed for **specific use cases** where **privacy, latency and efficiency are priorities** [56]. They can be fine-tuned at a **fraction of the cost** of an LLM using **accessible hardware** (24-48 GB GPUs), with deployments integrated into RAG pipelines or internal agents.

# Business Challenges

The development and integration of specialised language models offer enormous potential value, but they also pose a number of strategic, operational, and ethical challenges that must be addressed to achieve effective and sustainable adoption.

## ***Availability and Quality of Domain Data***

Specialised LLMs depend on specific, high-quality data. However, many organisations face barriers to collecting, cleaning, or structuring the necessary information. Key challenges include balancing the need for relevant data with the protection of sensitive information and corporate confidentiality.

## ***Governance, compliance, and information security***

The use of models that handle internal knowledge or regulated information requires high standards of governance. The following key challenges arise: ensuring traceability, access control, and compliance with regulations such as GDPR or ISO 27001, especially when models interact with personal or confidential data.

## ***Maintenance and updating of specialised knowledge***

A domain model can quickly become obsolete if it is not updated with new regulatory frameworks, technical criteria, or industry changes. Some of the key challenges are establishing continuous processes for updating the model and the associated knowledge base.

## ***Integration with existing systems and workflows***

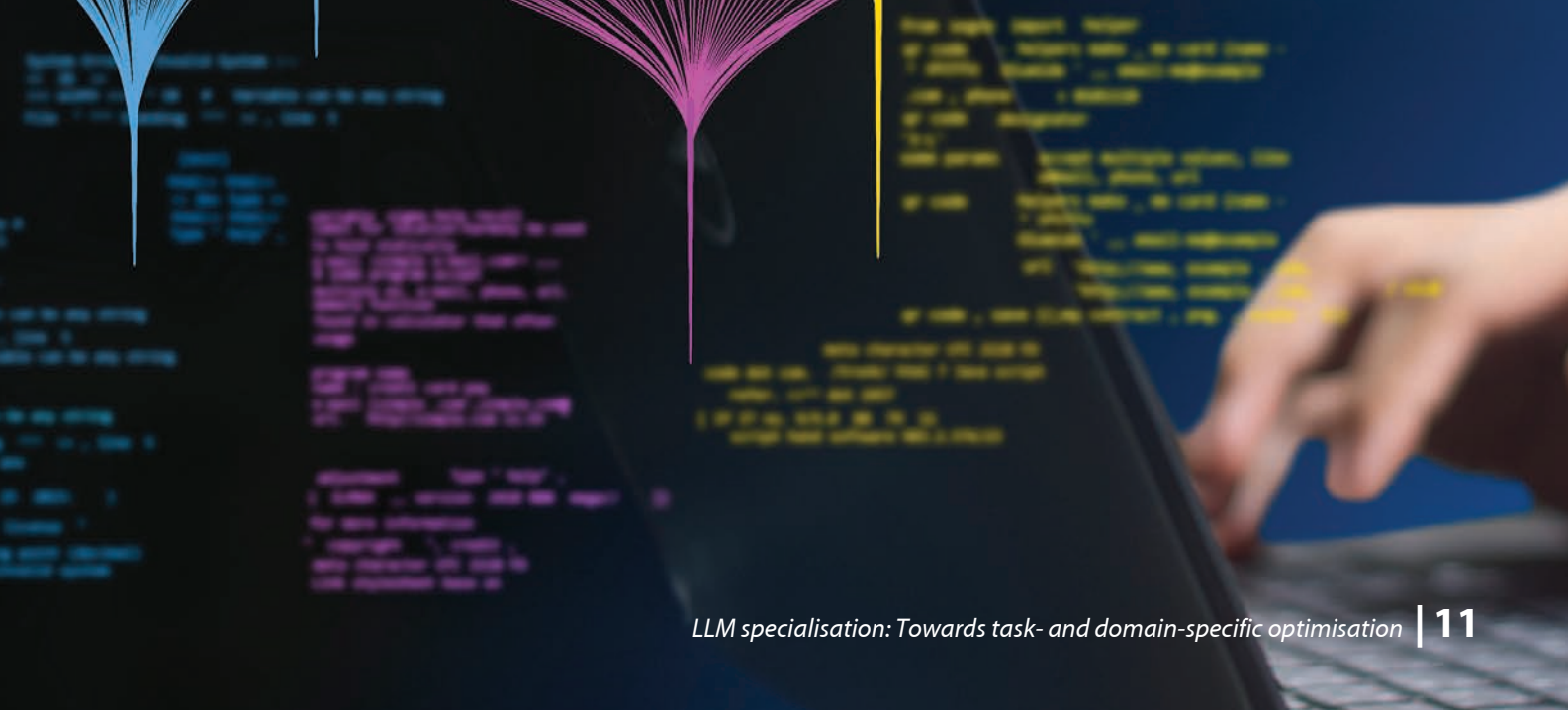
Adopting a specialised LLM is not just about the model, but also about its integration into the organisation's digital ecosystem (ERP, CRM, intranet, document management systems, etc.). The main challenge is to define an architecture that allows the model to be integrated without generating operational friction or duplication.

## ***Scalability and operation cost***

Customisation, fine-tuning and maintenance of models may require costly infrastructure or dependence on third parties. The main challenges are evaluating the return on investment and defining an efficient scaling strategy — for example, combining specialised LLMs with modular agents or off-the-shelf solutions.

## ***Balance between customisation and vendor dependency***

The current market offers numerous proprietary platforms (Copilot, Vertex AI, Claude, etc.) alongside open options (Llama, Mistral, etc.). Key challenges include finding the right balance between leveraging mature commercial capabilities and maintaining sovereignty over data and models.



# Conclusions

**Specialisation** represents one of the natural stages in the maturation of artificial intelligence within **production environments**. General-purpose language models remain extraordinarily versatile tools, ideal for rapid experimentation and covering a wide range of tasks, but their **performance** and **scalability** tend to decline when a high degree of **accuracy, traceability, and contextual understanding of the domain** is required.

In this scenario, **context engineering** strategies (such as ICL and, especially, RAG) and **efficient parameter tuning** techniques (PEFT using LoRA, QLoRA or adapters) are the most widely adopted. The optimal choice among these options does not lie in prioritising one over another, but in designing **hybrid architectures** where each technique contributes its specific value: RAG to incorporate updated knowledge with transparency and verifiability; PEFT to consolidate specialised reasoning, normative alignment and style; and ICL to facilitate agile experimentation or adjust specific behaviours without requiring complex deployments.

Furthermore, the **standardisation of tool use**, driven by initiatives such as the Model Context Protocol (MCP), is a decisive step towards expanding the capabilities of LLMs, allowing them to interact with external sources and access real-time information in a controlled manner. This functional integration not only increases their practical usefulness, but also lays the foundation for the development of **Multi Agent Systems (MAS)** that can improve processes such as RAG itself.

Also noteworthy is the evolution of concepts such as **Small Language Models (SLMs)** and **model routers**, which introduce a **right-sizing** approach that makes the transition to production of AI systems more efficient.

Also noteworthy is the growth of **integrated platforms**, which reduces the technical barriers to adoption, democratising the use of specialised LLMs, although it poses new challenges in terms of governance, transparency and data sovereignty. The real competitive advantage will lie not only in access to the model, but in the ability to strategically orchestrate its **integration, specialisation and continuous monitoring**.

In summary, the specialisation of language models marks the transition from exploration to consolidation of artificial intelligence as critical infrastructure in productive environments. The future of LLMs will depend not only on their **size or power**, but also on their ability to **integrate** intelligently with **specific data, tools and contexts**. The strategic combination of approaches such as RAG, PEFT and ICL, together with the standardisation of tool use and the rise of hybrid and scalable architectures, will enable the design of more efficient, auditable systems that are tailored to the real needs of each domain. Ultimately, competitiveness will be defined by the maturity with which organisations manage this ecosystem: how they govern their models, how they ensure their traceability, and how they align their technical evolution with ethical and digital sovereignty principles.



## Authors

Ernestina Menasalvas (UPM)  
Manuel Ángel Guzmán (Management Solutions)  
Sergio Ruiz (Management Solutions)  
Joaquín Velarde (Management Solutions)  
Daniel Rodríguez (Management Solutions)

# Bibliography

- [1] E. Mollick, «Scaling: The State of Play in AI,» One Useful Thing, September 2024. [En línea]. Available: <https://www.oneusefulthing.org/p/scaling-the-state-of-play-in-ai>.
- [2] A. McConnon, «Are bigger language models always better?,» IBM, [En línea]. Available: <https://www.ibm.com/think/insights/are-bigger-language-models-better>.
- [3] «ChatGPT vs Claude vs Gemini: ¿Cuál elegir en 2025 para proyectos de IA?,» DatiLab, September 2025. [En línea]. Available: <https://datilab.com/blog/chatgpt-vs-claude-vs-gemini-cual-elegir-2025-proyectos-ia>.
- [4] L. Wei, Z. Ying, M. He, Y. Chen, Q. Yang, Y. Hong, J. Lu, K. Zheng, S. Zhang, X. Li, W. Huang y Y. Chen, «Diabetica: Adapting Large Language Model to Enhance Multiple Medical Tasks in Diabetes Care and Management,» SCI-FM, ICLR 2025, 10.48550/arXiv.2409.13191, 2025.
- [5] H. Bansal, A. Hosseini, R. Agarwal, V. Q. T. Kazemi y Mehran, «Smaller, Weaker, Yet Better: Training LLM Reasoners via Compute-Optimal Sampling,» Google DeepMind, 2024.
- [6] Y. Lu, B. Yao, S. Zhang, Y. Wang, P. Zhang, T. Lu, T. J.-J. Li y D. Wang, «Human Still Wins over LLM: An Empirical Study of Active Learning on Domain-Specific Annotation Tasks,» 2023.
- [7] D. Anh-Hoang, V. Tran y L.-M. Nguyen, «Survey and analysis of hallucinations in large language models: attribution to prompting strategies or model behavior,» Front. Artif. Intell., 2025.
- [8] D. R. Rivera, «Evitar las alucinaciones de la GenAI,» VinculoTic, August 2025. [En línea]. Available: <https://vinculotic.com/salud/evitar-las-alucinaciones/>.
- [9] X.-K. Wu, M. Chen, W. Li, R. Wang, L. Lu, J. Liu, K. Hwang, Y. Hao, Y. Pan, Q. Meng, K. Huang, L. Hu, M. Guizani, N. Chao, G. Fortino, F. Lin, Y. Tian y D. Niyato, «LLM Fine-Tuning: Concepts, Opportunities, and Challenges,» Big Data and Cognitive Computing., 2025.
- [10] Q. Dong, L. Li, D. Dai, C. Zheng, J. Ma, R. Li, H. Xia, J. Xu, Z. Wu, T. Liu, B. Chang, X. Sun, L. Li y Z. Sui, «A Survey on In-context Learning,» arXiv, 2023.
- [11] W. Fan, Y. Ding, L. Ning, S. Wang, H. Li, D. Yin, T.-S. Chua y Q. Li, «A Survey on RAG Meeting LLM: Towards Retrieval-Augmented Large Language Models,» arXiv, 2024.
- [12] H. Chang, J. Park, S. Ye, S. Yang, Y. Seo, D.-S. Chang y M. Seo, «How Do Large Language Models Acquire Factual Knowledge During Pretraining?,» arXiv, 2024.
- [13] K. Buchholz, «The Extreme Cost Of Training AI Models,» Forbes, August 2024. [En línea]. Available: <https://www.forbes.com/sites/katharinabuchholz/2024/08/23/the-extreme-cost-of-training-ai-models/>.
- [14] D. Reigns, «Machine Learning Model Training Cost Statistics,» About Chromebooks, September 2025. [En línea]. Available: <https://www.aboutchromebooks.com/machine-learning-model-training-cost-statistics/>.
- [15] N. J. Prottasha, U. R. Chowdhury, S. Mohanto, T. Nuzhat, A. A. Sami, M. S. Ali, M. S. I. Sobuj, H. Raman, M. Kowsher y O. O. Garibay, «PEFT A2Z: Parameter-Efficient Fine-Tuning Survey for Large Language and Vision Models,» arXiv:2504.14117, 2025.
- [16] Y. Zhai, S. Tong, X. Li, M. Cai, Q. Qu, Y. J. Lee y Y. Ma, «Investigating the Catastrophic Forgetting in Multimodal Large Language Model Fine-Tuning,» Proceedings of Machine Learning Research, 2024.
- [17] Y. Luo, Z. Yang, F. Meng, Y. Li y J. Z. a. Y. Zhang, «An Empirical Study of Catastrophic Forgetting in Large Language Models During Continual Fine-Tuning,» IEEE Transactions on Audio, Speech and Language Processing, vol. 33, pp. 3776-3786, 2025.
- [18] Z. Han, C. Gao, J. Liu, J. Zhang y S. Q. Zhang, «Parameter-Efficient Fine-Tuning for Large Models: A Comprehensive Survey,» arXiv:2403.14608, 2024.
- [19] E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang y W. Chen, «LoRA: Low-Rank Adaptation of Large Language Models,» arXiv:2106.09685, 2021.
- [20] T. Dettmers, A. Pagnoni, A. Holtzman y L. Zettlemoyer, «QLoRA: Efficient Finetuning of Quantized LLM,» NeurIPS Proceedings, 2023.
- [21] Z. Hu, L. Wang, Y. Lan, W. Xu, E.-P. Lim, L. Bing, X. Xu, S. Poria y R. K.-W. Lee, «LLM-Adapters: An Adapter Family for Parameter-Efficient Fine-Tuning of Large Language Models,» EMNLP , 2023.
- [22] K. Chen, Y. Pang y Z. Yang, «Parameter-Efficient Fine-Tuning With Adapters,» arXiv:2405.05493, 2024.
- [23] «Low-rank adaptation (LoRA) fine tuning,» IBM, October 2025. [En línea]. Available: <https://dataplatfrom.cloud.ibm.com/docs/content/wsj/analyze-data/fm-tuning-methods-lora.html?context=wx&audience=wdp>.
- [24] «Recomendaciones de LoRA y QLoRA para LLM,» Google Cloud, October 2025. [En línea]. Available: <https://docs.cloud.google.com/vertex-ai/generative-ai/docs/model-garden/lora-qlora?hl=es-419>.
- [25] «¿Qué es la adaptación de bajo rango (LoRA)?,» Cloudflare, [En línea]. Available: [https://www.cloudflare.com/es-es/learning/ai/what-is-lora/#:~:text=La%20adaptaci%C3%B3n%20de%20bajo%20rango%20\(LoRA\)%20es%20un%20m%C3%A9todo%20para,personalizar%20modelos%20para%20contextos%20espec%C3%ADficos..](https://www.cloudflare.com/es-es/learning/ai/what-is-lora/#:~:text=La%20adaptaci%C3%B3n%20de%20bajo%20rango%20(LoRA)%20es%20un%20m%C3%A9todo%20para,personalizar%20modelos%20para%20contextos%20espec%C3%ADficos..)
- [26] L. Mei, J. Yao, Y. Ge, Y. Wang, B. Bi, Y. Cai, J. Liu, M. Li, Z.-Z. Li, D. Zhang, C. Zhou, J. Mao, T. Xia, J. Guo y S. Liu, «A Survey of Context Engineering for Large Language Models,» arXiv:2507.13334, 2025.
- [27] E. Kjosbakken, «How To Significantly Enhance LLM by Leveraging Context Engineering,» Towards Data Science, July 2025. [En línea]. Available: <https://towardsdatascience.com/how-to-significantly-enhance-llm-by-leveraging-context-engineering-2/>.
- [28] A. Zeichick, «¿Qué es la generación aumentada de recuperación (RAG)?,» OCl, September 2023. [En línea]. Available: <https://www.oracle.com/es/artificial-intelligence/generative-ai/retrieval-augmented-generation-rag/>.
- [29] J. Varughese, «¿Qué es el aprendizaje en contexto?,» IBM, [En línea]. Available: <https://www.ibm.com/es-es/think/topics/in-context-learning>. [Último acceso: November 2025].
- [30] X. Amatriain, «Prompt Design and Engineering: Introduction and Advanced Methods,» arXiv:2401.14423, 2024.
- [31] T. Feng, Y. Shen y J. You, «GraphRouter: A Graph-based Router for LLM Selections,» arXiv:2410.03834, 2024.
- [32] H. Han, Y. Wang, H. Shomer, K. Guo, J. Ding, Y. Lei, M. Halappanavar, R. A. Rossi, S. Mukherjee, X. Tang, Q. He, Z. Hua, B. Long, T. Zhao, N. Shah, A. Javari, Y. Xia y J. Tang, «Retrieval-Augmented Generation with Graphs (GraphRAG),» arXiv:2501.00309, 2024.
- [33] Y. Zhuang, C. Singh, L. Liu, J. Shang y J. Gao, «Vector-ICL: In-context Learning with Continuous Vector Representations,» arXiv:2410.05629, 2024.

- [34] C. Highmore, «In-Context Learning in Large Language Models: A Comprehensive Survey.» 10.20944/preprints202407.0926.v1. , 2024.
- [35] Z. Li, Z. Xu, L. Han, Y. Gao, S. Wen, D. Liu, H. Wang y D. N. Metaxas, «Implicit In-context Learning.» arXiv:2405.14660, 2024.
- [36] B. Sarmah, D. Mehta, B. Hall, R. Rao, S. Patel y S. Pasquali, «HybridRAG: Integrating Knowledge Graphs and Vector Retrieval Augmented Generation for Efficient Information Extraction.» In Proceedings of the 5th ACM International Conference on AI in Finance (ICAIF '24), 2024.
- [37] X. Hou, Y. Zhao, S. Wang y H. Wang, «Model Context Protocol (MCP): Landscape, Security Threats, and Future Research Directions.» arXiv:2503.23278, 2025.
- R. MacManus, «No, MCP Hasn't Killed RAG — in Fact, They're Complementary.» The New Stack, May 2025. [En línea]. Available: <https://thenewstack.io/no-mcp-hasnt-killed-rag-in-fact-theyre-complementary/>.
- [38] C. Vidal, «RAFT: A new way to teach LLM to be better at RAG.» Azure AI Foundry Blog, March 2024. [En línea]. Available: <https://techcommunity.microsoft.com/blog/azure-ai-foundry-blog/raft-a-new-way-to-teach-llm-to-be-better-at-rag/4084674>.
- [39] S. Schürch, «How to Make Your LLM More Accurate with RAG & Fine-Tuning.» Towards Data Science, March 2025. [En línea]. Available: <https://towardsdatascience.com/how-to-make-your-llm-more-accurate-with-rag-fine-tuning/>.
- [40] «In-Context Learning: Extreme vs. Fine-Tuning, RAG.» Meta Quantum Today, May 2024. [En línea]. Available: <https://meta-quantum.today/?p=2990>.
- [41] O. Kamath, «Main Page.» MeetCody.AI, [En línea]. Available: <https://meetcody.ai/es/blog/rag-como-servicio-desbloquea-la-ia-generativa-para-tu-empresa/>. [Último acceso: November 2025].
- [42] «RAG vs Fine-Tuning for LLM: A Comprehensive Guide with Examples.» Hugging Face, August 2024. [En línea]. Available: <https://huggingface.co/blog/airabbitX/rag-vs-fine-tuning-for-llm-a-com>.
- [43] J. Ferrer, «Fine-Tuning LLM: A Guide With Examples.» Data Camp, December 2024. [En línea]. Available: <https://www.datacamp.com/tutorial/fine-tuning-large-language-models>.
- [44] «Model optimization.» OpenAI, [En línea]. Available: <https://platform.openai.com/docs/guides/model-optimization>. [Último acceso: November 2025].
- [45] Z. Gekhman, G. Yona, R. Aharoni, M. Eyal, A. Feder, R. Reichart y J. Herzig, «Does Fine-Tuning LLM on New Knowledge Encourage Hallucinations?» EMNLP 2024, 2024.
- [46] O. Ovadia, M. Brief, M. Mishaeli y O. Elisha, «Fine-Tuning or Retrieval? Comparing Knowledge Injection in LLM.» arXiv:2312.05934v3 , 2023.
- [47] H. Soudani, E. Kanoulas y F. Hasibi, «Fine Tuning vs. Retrieval Augmented Generation for Less Popular Knowledge.» 12-22. 10.1145/3673791.3698415.
- [48] «RAG vs. fine-tuning: Choosing the right method for your LLM.» Super Annotate, August 2024. [En línea]. Available: <https://www.superannotate.com/blog/rag-vs-fine-tuning#:~:text=RAG%20is%20a%20good%20fit,that%20require%20specialized%2C%20precise%20responses..>
- [49] I. Mienye, N. Jere, G. Obaido, O. Ogunraku, Esenogho y C. Modisane, «Large language models: an overview of foundational architectures, recent trends, and a new taxonomy.» Discover Applied Sciences. 7. 10.1007/s42452-025-07668-w, 2025.
- [50] B. Gao, X. Wang, Y. Yang y D. Clifton, «Optimization-Inspired Few-Shot Adaptation for Large Language Models.» 10.48550/arXiv.2505.19107, 2025.
- [51] «Agentes para Microsoft 365 Copilot.» Microsoft, [En línea]. Available: <https://www.microsoft.com/es-es/microsoft-365-copilot/agents>. [Último acceso: November 2025].
- [52] «Managed Agents in Copilot Studio: Everything You Need to Know.» Global Sharepoint, October 2025. [En línea]. Available: <https://global-sharepoint.com/copilot/document-processors-managed-agent/>.
- [53] Alantra, «From restructuring to scalable, profitable growth.» Alantra, May 2025. [En línea]. Available: [https://media.expert.ai/expertai/uploads/2020/08/20250528\\_Alantra\\_ExpertAi\\_From-restructuring-to-scalable-profitable-growth-20250528.pdf?](https://media.expert.ai/expertai/uploads/2020/08/20250528_Alantra_ExpertAi_From-restructuring-to-scalable-profitable-growth-20250528.pdf?)
- [54] «Main Page.» ZFORT Group, [En línea]. Available: <https://www.zfort.com/>. [Último acceso: November 2025].
- [55] «Maximize Automation & Enhance Customer Interactions with LLM.» Scopic, [En línea]. Available: <https://scopicsoftware.com/llm-development-services/>. [Último acceso: November 2025].
- [56] Y. Zhang, X. Zhao, Z. Wang, G. Cheng, Y. Xu, S. Deng y J. Yin, «LightRouter: Towards Efficient LLM Collaboration with Minimal Overhead.» arXiv:2505.16221v1, 2025.
- [57] D. Stripelis, Z. Hu, J. Zhang, Z. Xu, A. D. Shah, H. Jin, Y. Yao, S. Avestimehr y C. He, «TensorOpera Router: A Multi-Model Router for Efficient LLM Inference.» EMNLP 2024, 2024.
- [58] T. Feng, Y. Shen y J. You, «GraphRouter: A Graph-based Router for LLM Selections.» 10.48550/arXiv.2410.03834. , 2024.
- [59] S. Masoudnia y R. Ebrahimpour, «Mixture of experts: a literature survey.» Artificial Intelligence Review. 42. 10.1007/s10462-012-9338-y. , 2014.
- [60] «Model router for Azure AI Foundry (preview).» Microsoft, 2025. [En línea]. Available: <https://learn.microsoft.com/en-us/azure/ai-foundry/openai/concepts/model-router>. [Último acceso: November 2025].
- [61] «Main page.» Storytell.ai, [En línea]. Available: <https://storytell.ai/>. [Último acceso: November 2025].
- [62] «NVIDIA AI-Blueprints.» Github, [En línea]. Available: <https://github.com/NVIDIA-AI-Blueprints/llm-router>. [Último acceso: November 2025].
- [63] S. Savvov, «Your Company Needs Small Language Models.» Towards Data Science, December 2024. [En línea]. Available: <https://towardsdatascience.com/your-company-needs-small-language-models-d0a223e0b6d9/>.
- [64] J. Ferrer, «Todo lo que sabemos sobre GPT-5.» Datacamp, February 2025. [En línea]. Available: <https://www.datacamp.com/es/blog/everything-we-know-about-gpt-5>.



**POLITÉCNICA**

UNIVERSIDAD  
POLITÉCNICA  
DE MADRID



The Universidad Politécnica de Madrid is a multi-sector and multi-disciplinary Public Law Entity, which carries out teaching, research and scientific and technological development activities.

[www.upm.es](http://www.upm.es)

Management Solutions is an international consulting firm, focused on business, finance, risk, organization, technology and process consulting, operating in more than 50 countries and with a team of more than 4,000 professionals working for more than 2,200 clients worldwide.

[www.managementsolutions.com](http://www.managementsolutions.com)

For more information visit

**[blogs.upm.es/catedra-idanae/](http://blogs.upm.es/catedra-idanae/)**