Appsin.Ai

SERVICE

Large Language Model Development: Custom, Fine-Tuned and Integrated LLMs

We select, adapt and deploy large language models for your tasks, from fine-tuned open-source models on your servers to integrations with hosted APIs.

https://appsin.ai · +91-9025555584 · sales@appsin.ai
AI development for businesses in India, the GCC and the USA

OVERVIEW

LLM Development

A large language model is the engine behind chat assistants, document tools and copilots. We help you pick the right one, adapt it to your language and tasks, and run it reliably inside your products. That might mean a fine-tuned open-source model on your own GPUs, or a hosted model wrapped in your rules and data.

Our LLM work covers model selection and benchmarking, fine-tuning on your examples with methods such as LoRA, retrieval over your documents, prompt design, and the serving layer that handles load, latency and cost. We work with transformer models including GPT, Llama, Mistral and Gemini, using PyTorch and the Hugging Face ecosystem.

Language models bring their own risks: bias, invented facts, harmful output and leaks of sensitive data. We test for them before launch with domain-specific evaluation sets and red-team prompts, and keep monitoring after go-live so quality does not slip as your data and the underlying models change.

BUILT FOR

- Product companies building on LLMs
- Enterprises that need a private, self-hosted model
- Teams working with legal, medical or financial language
- Businesses serving customers in several languages
- CTOs weighing hosted APIs against open-source models

FEATURES

What you get with LLM Development

01 **Model selection and benchmarking**
We test candidate models on your own tasks and compare accuracy, speed, cost per request and licence terms before you commit to one.

02 **Custom domain models**
Language models for specialised vocabulary in fields such as law, medicine or finance, built on an open-source base and trained on your corpus.

03 **Fine-tuning**
Full or parameter-efficient fine-tuning (LoRA, QLoRA) on curated examples, teaching a model your format, tone and terminology.

04 **Retrieval and enterprise search**
RAG pipelines with chunking, embeddings and re-ranking, so answers come from your current documents rather than the model's memory.

05 **LLM integration**
Models embedded in support platforms, content management systems, CRMs and internal tools through APIs, rolled out without halting daily operations.

06 **Multilingual applications**
Translation, multilingual chat and text generation, with quality evaluated separately in each language you serve.

07 **Self-hosted inference**
Open-source models served on your cloud GPUs or servers with inference engines such as vLLM, using batching and quantisation to control cost.

08 **Evaluation and red-teaming**
Automated and human-graded tests for accuracy, bias and safety, plus adversarial prompts that find failure cases before your users do.

09 **Monitoring and retraining**
Dashboards for usage, quality and cost, alerts on errors and drift, and retraining with updated data when performance falls.

WHY IT MATTERS

The difference it makes to your business

01 **A model that knows your domain**
Tuning on your documents and examples teaches the model the terms, abbreviations and conventions a general model gets wrong.

02 **Control over cost and latency**
A smaller tuned model can often handle a narrow task more cheaply and quickly than the largest general model. We measure this on your workload before recommending it.

03 **Data kept on your terms**
With a self-hosted model, prompts, documents and outputs never pass through a third-party API.

04 **No single-vendor dependence**
An abstraction layer and open-source options let you move between models as prices, licences and quality change.

HOW WE DELIVER

How we deliver LLM Development

01 **Research and design**
We define what the model must do, the quality bar and the constraints, then design the architecture: hosted API, fine-tuned model, RAG or a mix.

02 **Collect and prepare data**
Examples and documents are gathered from varied sources, cleaned of noise, duplicates and personal data, and checked for bias before training.

03 **Train and fine-tune**
We adapt the chosen base model with PyTorch on GPU infrastructure, logging each run so results can be reproduced and compared.

04 **Evaluate and test for safety**
The model is tested on domain tasks, bias and harmful-output checks and red-team prompts, and compared with the base model and alternatives.

05

Deploy and monitor

We deploy behind an API with logging, rate limits and dashboards, then track quality and retrain as data and requirements change.

FAQ

Questions about LLM Development

Do we need our own LLM, or can we use a hosted model like ChatGPT?

Many businesses do well with a hosted model plus retrieval over their own documents. Your own fine-tuned or self-hosted model makes sense when data cannot leave your network, when request volumes make API costs high, or when you need behaviour a hosted model will not reliably produce. We test both routes on your tasks before recommending one.

Which large language models do you work with?

Hosted models such as OpenAI GPT, Google Gemini and Anthropic Claude, and open-source models such as Llama and Mistral. We also build hybrid systems that combine an LLM with classic machine learning, for example a classifier that routes requests before the language model answers.

What does LLM development cost?

Cost depends on the base model and its size, the amount of training data to collect and clean, GPU time for training, hosting and serving, and the integration work. Using a hosted API with retrieval is usually cheapest to start; a self-hosted fine-tuned model costs more up front but can cost less per request at volume.

How much data is needed to fine-tune an LLM?

For a narrow task such as a fixed output format or a classification style, a few hundred to a few thousand high-quality examples is often a workable starting point. Quality matters far more than volume. Teaching a model a whole new domain takes much more text, and we tell you after a first trial whether your data is enough.

Can the model run on our own servers?

Yes. Open-source models can be hosted on your own GPUs or cloud account. We size the hardware to the model and expected traffic, and use quantisation where it keeps quality acceptable at lower cost.

How do you test an LLM for accuracy and bias?

We build an evaluation set from your real tasks with expected answers, score the model automatically and with human reviewers, run bias and harmful-output checks across sensitive topics, and try adversarial prompts. The same set is re-run after every change so you can see whether quality went up or down.

What ongoing support do you offer for LLMs?

Monitoring of outputs for errors, bias and drift, retraining with new data, prompt and retrieval adjustments, usage and cost dashboards, upgrades when better base models appear, and capacity planning as traffic grows.

NEXT STEP

Find the right language model for your use case

Book a free 30-minute call. Share your use case and data constraints, and we will suggest which models to benchmark, where to host them and whether fine-tuning is worth it.

Book a free consultation: https://appsin.ai/book-a-consultation/

WhatsApp or call: +91-9025555584

About AppsInAi

AppsInAi Private Limited is an AI development company in Coimbatore, India. We build generative AI assistants, AI agents, chatbots, machine learning models, computer vision and data platforms on our clients' own data, from a first proof of concept to production, with monitoring and support after launch.

Contact
sales@appsin.ai
+91-9025555584
Housing Unit, Sf No. 393, CMS School Backside, 3rd Street, Nehru Nagar, Ganapathy, Coimbatore 641006, India
https://appsin.ai