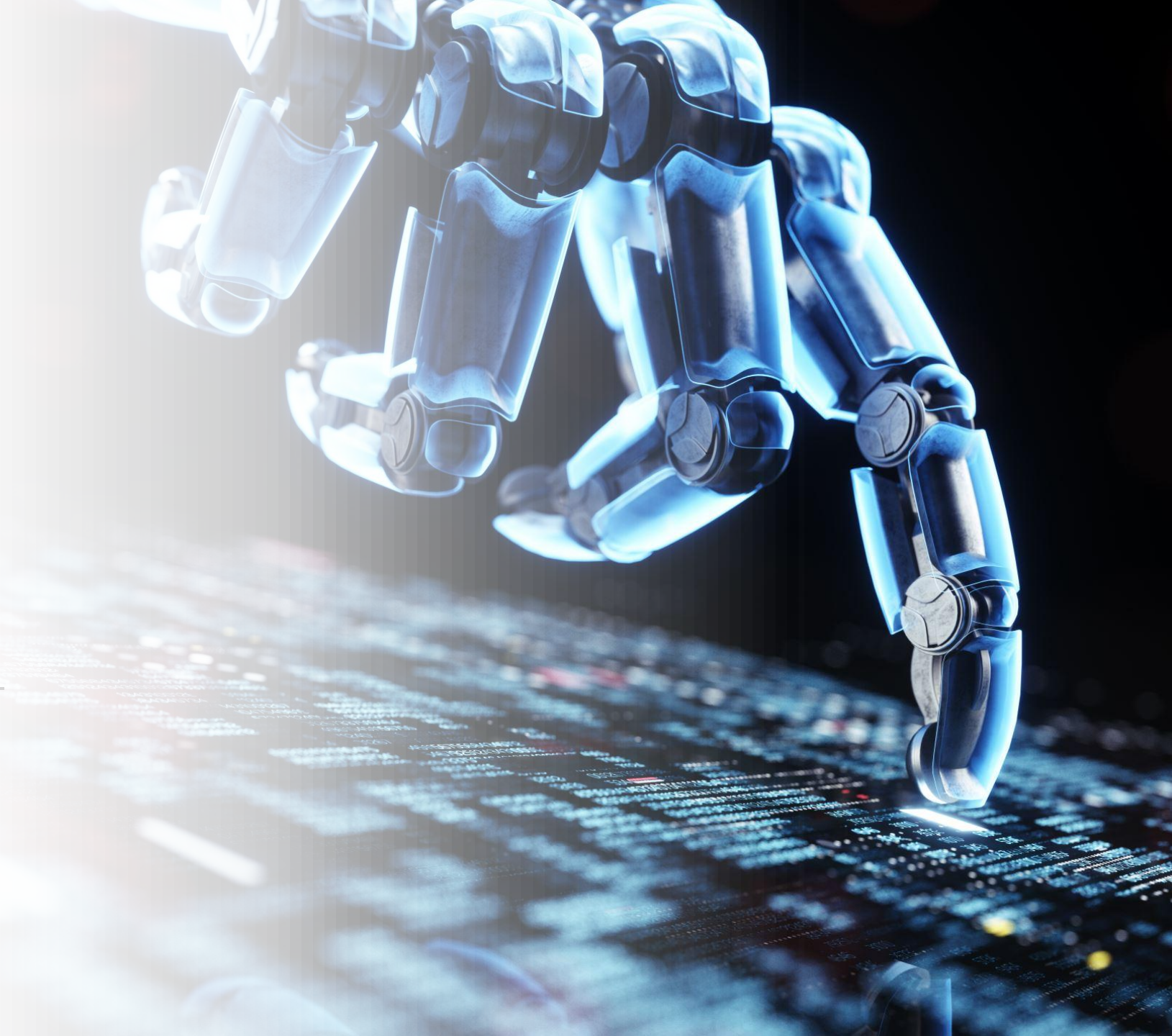
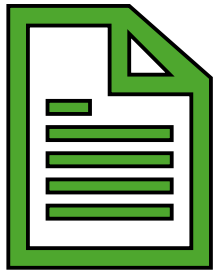
Nirmata

AI-Powered Automated Audit

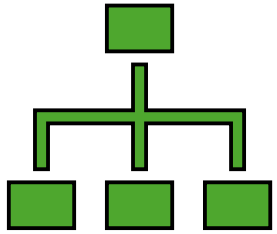


Project Overview



Purpose:

To automate the preparation, document review (Stage 1), on-site audit preparation (stage 2), and report writing processes of management system audits while maintaining manual oversight for critical decisions such as granting certifications.



Scope:

The tool will support internal audit processes, document management, and report generation, streamlining the workflow and improving efficiency.

Audit Process

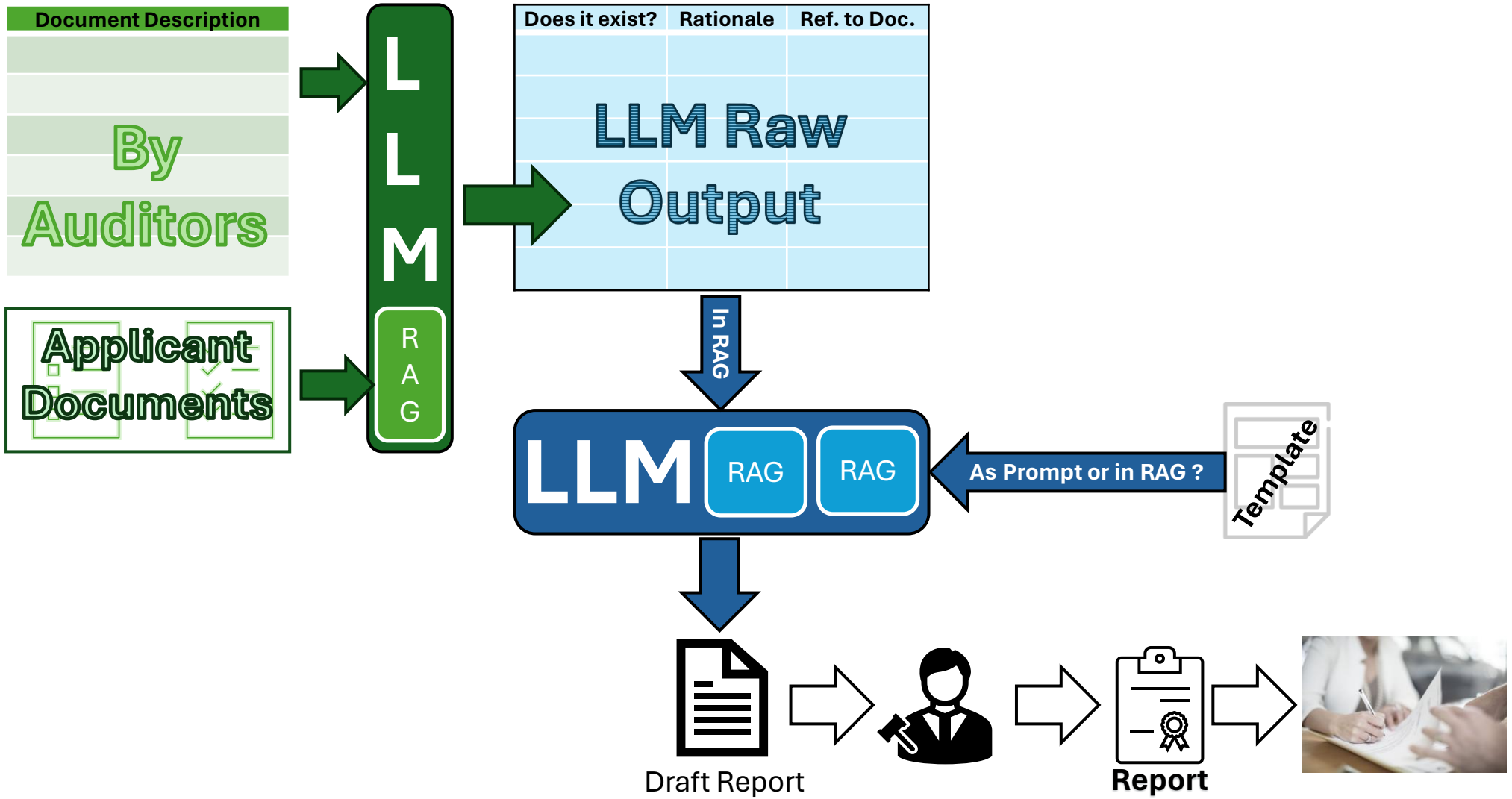
Different Tasks of LLM

Assess compliance of given documents w.r.t a set of requirements

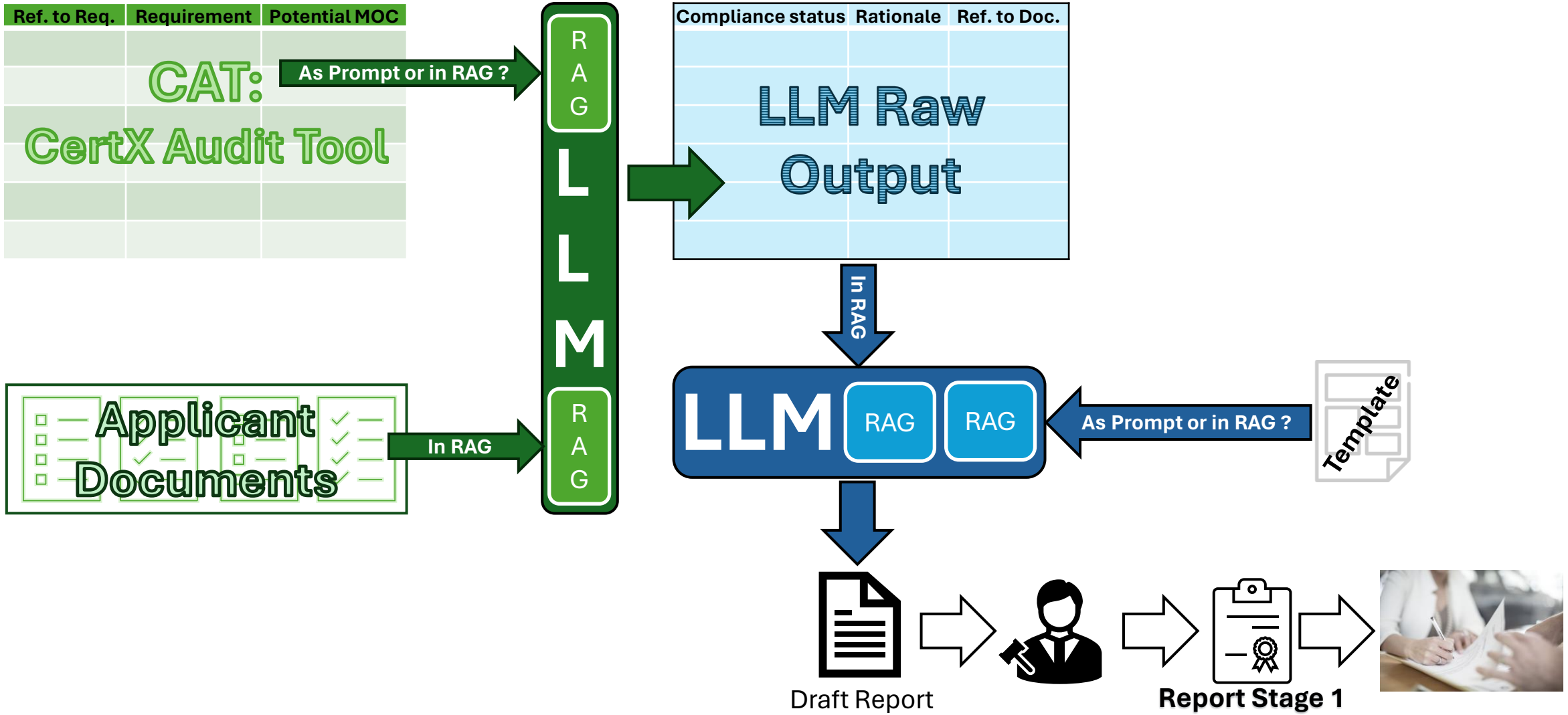
Creates a report following a template by polishing the result of assessment

Assists in finding questions/points to be raised given a set of documents

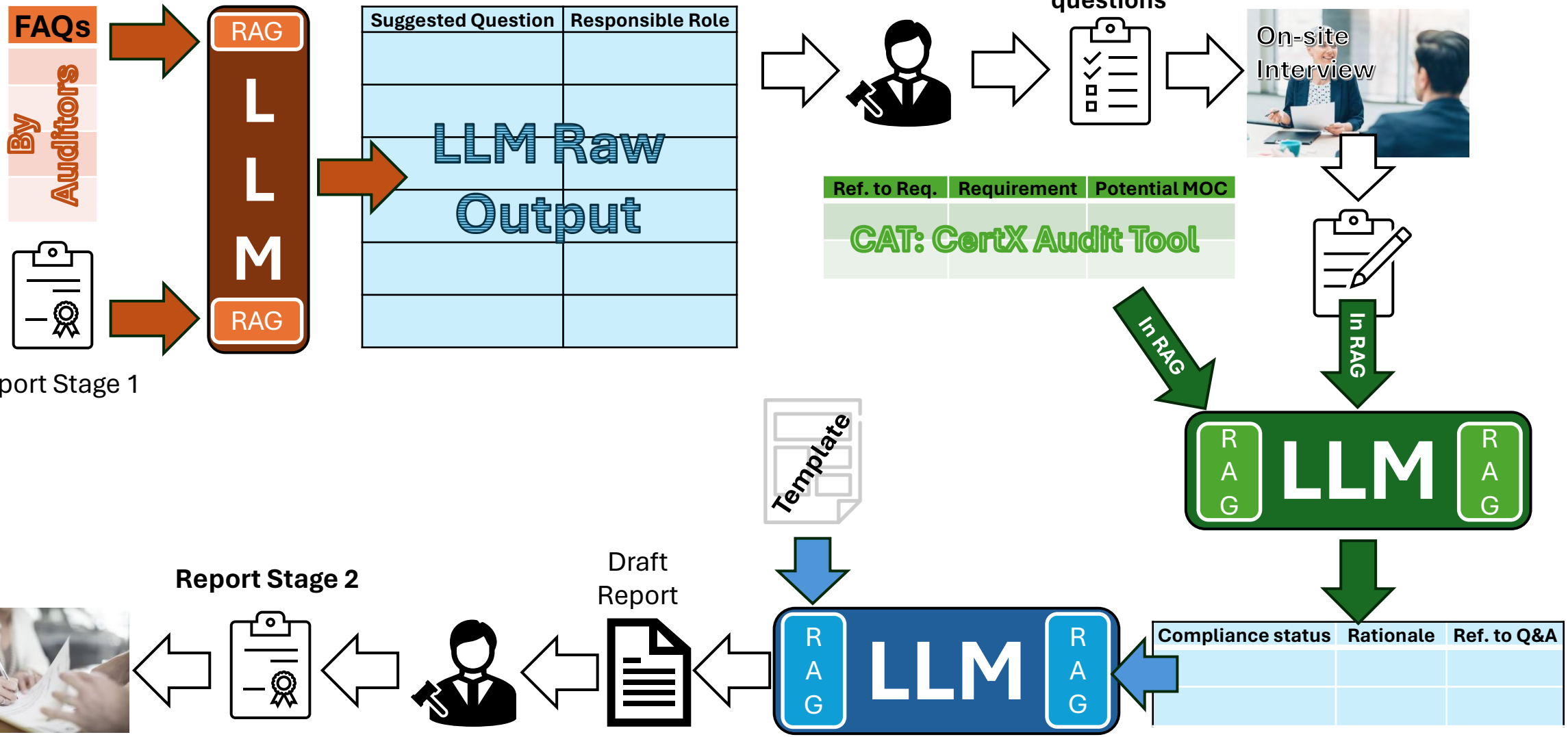
Preparation



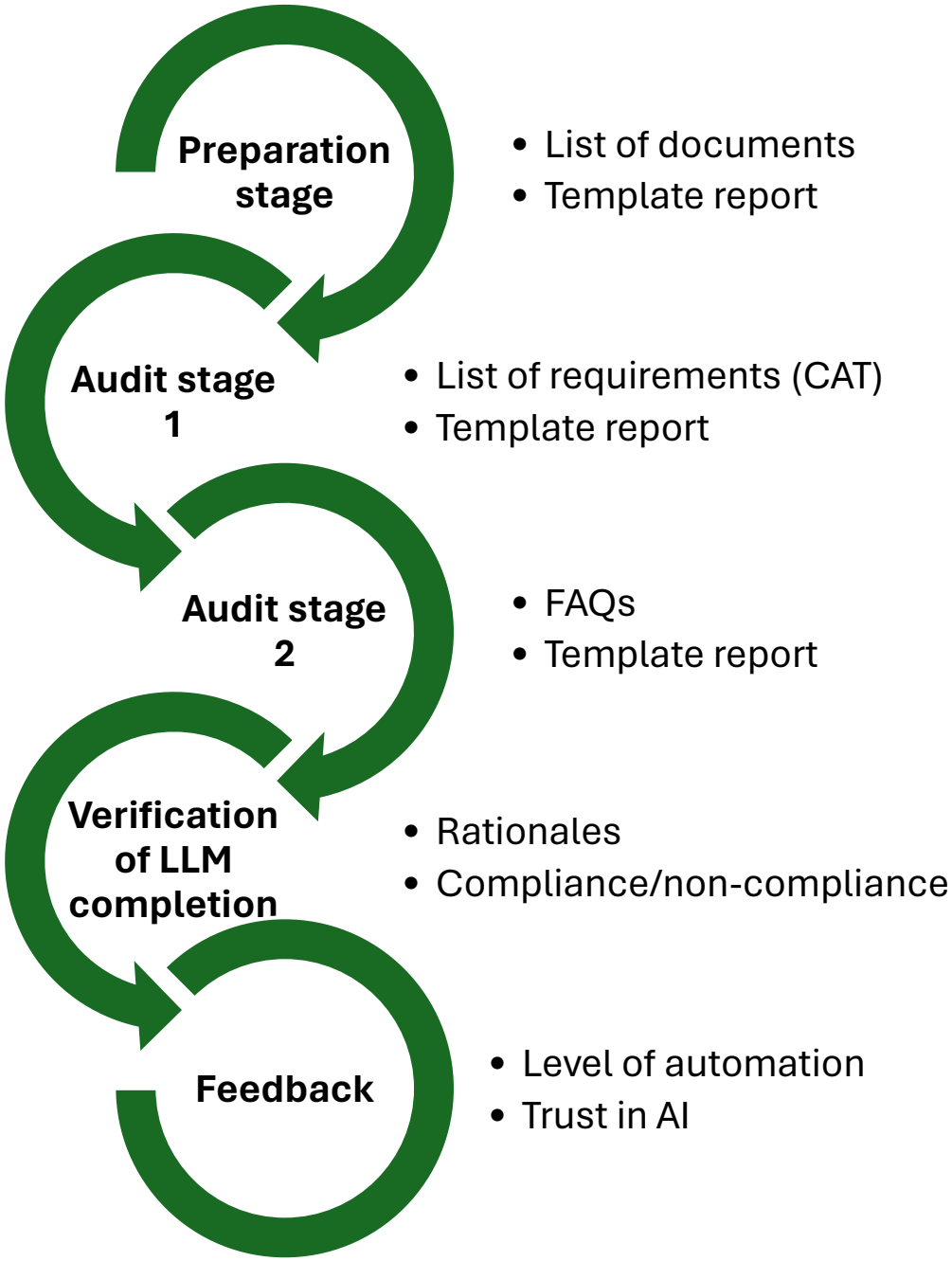
LLM in Stage 1



LLM in Audit Stage 2



Where Auditors Help



Quantifiable Goals

Efficiency Improvement: Reduce the time spent on audit preparation, document review, and report writing

by 50%

Accuracy Enhancement: Accurately identifying compliance and non-compliance issues within documents.

> 95%

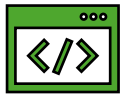
Report Standardization: Consistency in report formatting and content structure across all audits.

> 99.9%

Functional Requirements

Audit Preparation	• Automate the creation of audit checklists based on the specific standards and criteria applicable to the auditee. • Generate a list of required documents and evidence based on the scope of the audit.
Document Review	• Automatically analyze provided documents for completeness, relevance, and compliance with the applicable standards. • Identify and highlight areas of non-compliance, potential risks, and areas for improvement.
Points of interest and questions	• Generate list of questions and issues to be used during the on-site interview
Report Writing	• Generate detailed audit reports summarizing findings, non-compliances, and recommendations for Stage 1 and Stage 2 audits. • Customize report templates to include audit scope, methodology, findings, conclusions, and recommendations.
Integration and Data Handling	• Ensure seamless integration with existing document management systems and databases. • Implement robust data extraction, processing, and summarization capabilities for analyzing audit evidence.

Non-Functional Requirements



Usability

User-friendly interface for auditors to input data, review automated analyses, and edit reports.

Comprehensive training materials and support documentation.

Reliability

Ensure the tool operates with a 99% uptime, excluding scheduled maintenance.

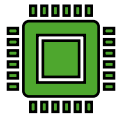
Implement error handling and recovery processes to manage failures gracefully.

Security

Enforce strict access controls and authentication mechanisms to protect sensitive audit information.

Comply with data protection regulations (e.g., GDPR) for handling personal and sensitive business information.

DECISION: On-premises



Scalability

Design the system to easily accommodate an increasing number of audits without degradation in performance.

Performance

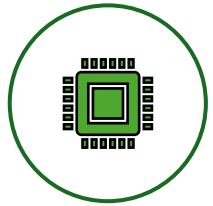
Ensure the system can process documents and generate reports within acceptable timeframes (e.g., less than 5 minutes for document analysis).

Tools and Technologies

AI AND MACHINE LEARNING: GPT AND RAG MODELS FOR NATURAL LANGUAGE UNDERSTANDING, DOCUMENT ANALYSIS, AND CONTENT GENERATION.

SOFTWARE DEVELOPMENT: PROGRAMMING LANGUAGES (E.G., PYTHON), FRAMEWORKS, AND LIBRARIES SUITABLE FOR AI MODEL INTEGRATION AND WEB APPLICATION DEVELOPMENT.

DATABASE MANAGEMENT: SECURE AND SCALABLE DATABASE SOLUTIONS FOR STORING AUDIT RECORDS, REPORTS, AND EVIDENCE.



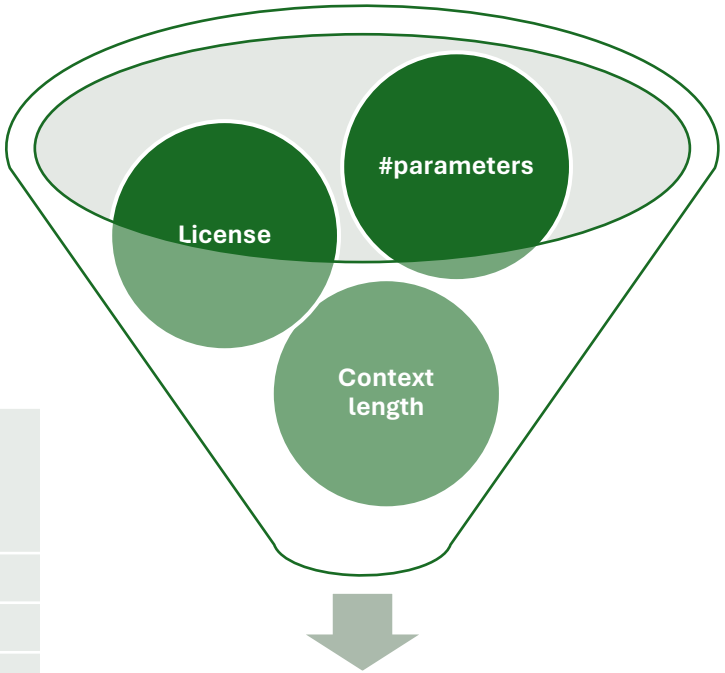
INTEGRATION TOOLS: APIS AND MIDDLEWARE FOR CONNECTING WITH EXISTING DOCUMENT MANAGEMENT SYSTEMS AND DATABASES.

SECURITY: TOOLS AND SERVICES FOR IMPLEMENTING AUTHENTICATION, AUTHORIZATION, ENCRYPTION, AND DATA PROTECTION.

INFRASTRUCTURE: ON-PREMISES SERVERS FOR HOSTING THE APPLICATION, ENSURING RELIABILITY, AND SCALABILITY.

CLOUD: FOR ENCRYPTED MODELS PARAMS AND DATA

Available LLMs



Language Model	Params (B)	Avg. Open LLM Leaderboard	Context Length	Licence
MPT-7B	7	NA	84k (ALiBi)	Apache 2.0, CC BY-SA-3.0
UL2	20	NA	512, 2048	Apache 2.0
Mistral 7B	7	54.96	4096-16K	Apache 2.0
MPT-30B	30	NA	8192	Apache 2.0, CC BY-SA-3.0
StableLM-Alpha	3 – 65	46.58	4096	CC BY-SA-4.0
LLaMA 2	3 – 70	68.24	4096	Custom Free if you have under 700M users and you cannot use LLaMA outputs to train other LLMs besides LLaMA and its derivatives
OpenHermes	7, 13	71.38	4096	MIT
SOLAR	10.7	66.04	4096	apache-2.0

Need for Fine-tuning?

Domain-Specific Data

Past audit reports

Regulatory texts

Compliance guidelines

Challenges

Substantial Dataset

Representative Dataset

Multiple Tasks

From Pre-trained LLMs to Fine-tuning

Exhaustive testing

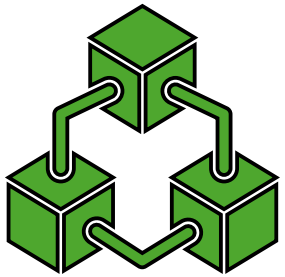
Prompt engineering

One-shot and even few-shot inference

Other heuristic SW engineering for pre- and post-processing

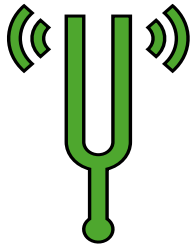
Performance requirement not met?

Need for fine-tuning



Pre-trained models: For tasks involving general language understanding and generation pre-trained models might suffice.

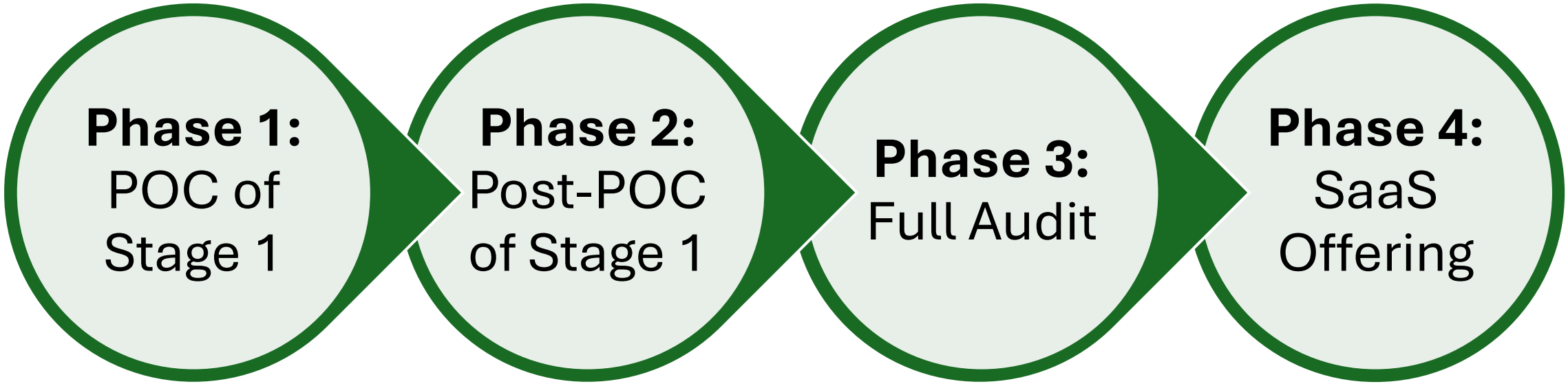
Example: generating standard audit report sections



Fine-tuning: For specialized tasks unique to a domain fine-tuning with domain-specific data could significantly improve performance.

Example: interpreting technical standards or regulations specific to an industry

Next Steps – Tentative Deadlines



- POC of ISO 17021 on CertX MM (Mid-April'24) – 1st LLM

- POC of ISO 26262 – 1st LLM (Aug. 2024)
- POC of ISO 26262 – 2nd LLM (Nov. 2024)
- Release for 2024 on Dec. 2024

- Assuming reusability of LLMs in stage 1:
  - Release of stage 2 in Summer 2025

- End of 2025

Project Plan – Phase 1

- **POC** on ISO 17021 (CertX MM) by mid-April (tentative)
- **Scope**: Audit stage 1 – Document Review (1st LLM)

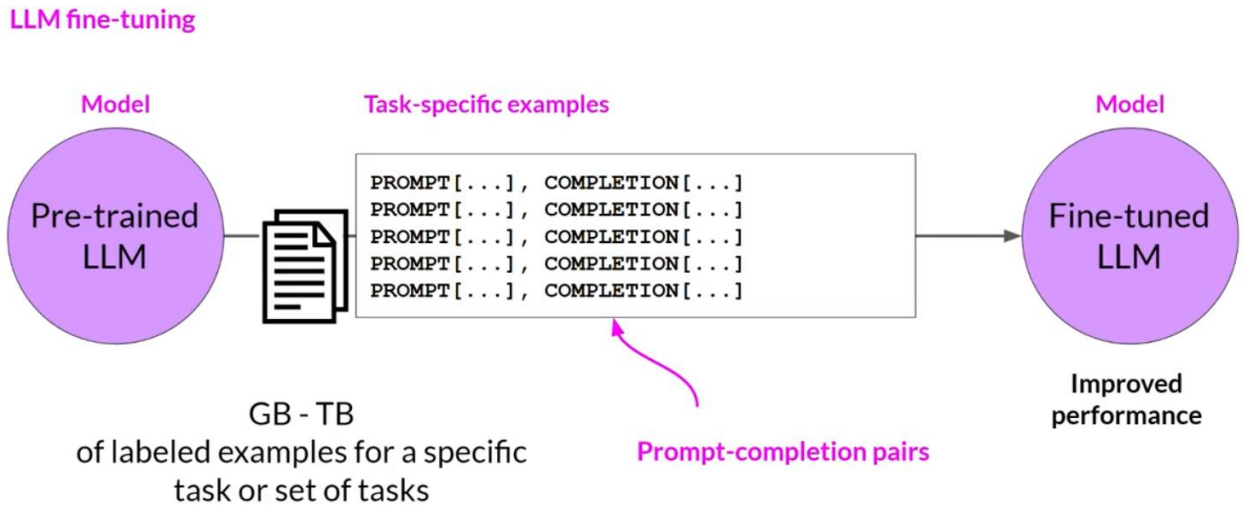
Fine-Tuning

Some technical details

Instruction Fine-tuning

Adapt a pre-trained LLM to perform better on tasks specified by instruction prompts.

- **Training Data Preparation**: Utilize publicly available or proprietary datasets, possibly reformatted with instruction prompts using prompt template libraries, to create a dataset suitable for fine-tuning.
- **Dataset Splits:** Divide the dataset into training, validation, and test splits

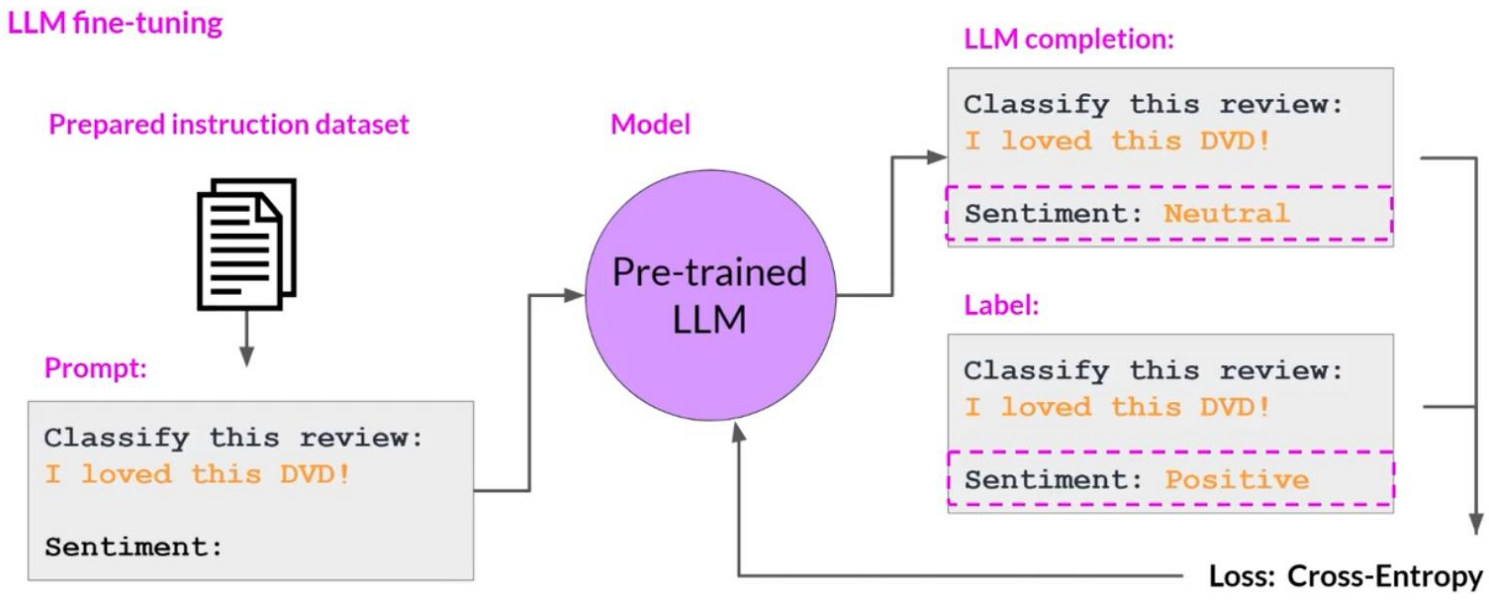


Fine-tuning Process

Training: The model generates completions for selected prompts from the training dataset, which are then compared to the expected responses to calculate the loss.

Loss Calculation: Use cross-entropy to measure the difference between the model's output token distribution and the expected token distribution from the training label.

Weight Updates: Adjust the model's weights through backpropagation based on the calculated loss to improve task-specific performance.



Parameter Efficient Fine-Tuning (PEFT)

- The same original model can be trained for other tasks individually while only training a small (new) subset of parameters

- LoRA (Low-Rank Adaptation)
- Additive methods
  - Adapters
  - Soft prompts (prompt tuning ≠ prompt engineering)