

# A Hybrid Multi-Modal AI Orchestration Framework for Scalable Agricultural Decision Support Systems

1<sup>st</sup> Keerthana K V  
Department of Computer Science  
and Engineering,  
JSS Science and Technology  
University, Mysuru, India  
Email: keerthanakv08@gmail.com

2<sup>nd</sup> Prajwal M  
Department of Computer Science and  
Engineering,  
JSS Science and Technology  
University, Mysuru, India  
Email: prajwalm0105204@gmail.com

3<sup>rd</sup> H Amit Kumar  
Department of Computer Science  
and Engineering,  
JSS Science and Technology  
University, Mysuru, India  
Email: amith368@gmail.com

4<sup>th</sup> Deepak C Yadav  
Department of Computer Science  
and Engineering,  
JSS Science and Technology  
University, Mysuru, India  
Email: sedepakayadav@gmail.com

5<sup>th</sup> Raksha R  
Department of Computer Science and  
Engineering,  
JSS Science and Technology  
University, Mysuru, India  
Email: raksha@jssstuniv.in

**Abstract**—Agricultural decision-making in real-world scenarios often requires combining multiple types of information including crop conditions, disease symptoms, and market trends. However, most existing systems address these tasks independently, which limits their practical usability. In this work, we design and implement a hybrid multi-modal AI orchestration framework as part of the AgroSync platform to provide unified agricultural decision support. The proposed system integrates three key components: a language-based advisory module, a vision-based crop disease analysis module, and a statistical forecasting module based on ARIMA. Instead of treating these components as separate solutions, we introduce an orchestration pipeline that manages request classification, contextual data injection, model selection, and structured response generation. The architecture follows a microservice-based design to ensure modularity and scalability, making it suitable for deployment in resource constrained rural environments. The forecasting module uses an ARIMA(5,1,0) configuration to analyze commodity price trends, while the advisory module provides contextual recommendations using language models. Experimental evaluation focuses on system-level behavior rather than isolated model accuracy. Results show that the orchestration pipeline maintains consistent output structure, reliable integration across modules, and stable execution of forecasting workflows. The proposed approach demonstrates how combining multiple AI paradigms within a single coordinated system can improve the practicality of agricultural decision support solutions.

**Keywords**—Artificial Intelligence; Multi-Modal AI; Agricultural Decision Support Systems; AI Orchestration; Crop Disease Detection; ARIMA Forecasting

## I. INTRODUCTION

Agriculture in developing regions often involves decision-making under uncertainty, where farmers need to consider multiple factors such as soil conditions, crop health, weather patterns, and market prices. While several digital tools have been developed for tasks like crop recommendation, disease detection, and price forecasting, these systems usually operate

independently and do not provide a unified decision-making experience.

In practical scenarios, these tasks are interconnected. For example, identifying a crop disease should ideally lead to immediate treatment suggestions, while market trends should influence decisions on when to sell produce. However, most existing agricultural AI solutions focus on optimizing individual models rather than integrating them into a single coordinated system.

In this work, we focus on addressing this gap by designing a hybrid multi-modal AI orchestration framework within the AgroSync platform. The goal is not to build a new standalone model, but to create a system that effectively combines multiple AI services into a structured pipeline. The framework integrates language-based advisory reasoning, image-based crop disease analysis, and time-series forecasting using ARIMA models.

A key contribution of this work is the introduction of a deterministic orchestration pipeline that manages how user requests are processed. This includes classifying the type of request, injecting relevant contextual information such as crop type and region, selecting the appropriate AI module, and enforcing a structured output format. This approach ensures consistency across responses and reduces ambiguity when multiple AI services are involved.

The proposed architecture follows a microservice-based design, allowing each AI component to function independently while still being part of a unified system. This makes the framework scalable and adaptable for real-world deployment, especially in environments with limited computational resources.

Overall, this work emphasizes system-level integration over isolated model performance, highlighting the importance of orchestration in building practical and deployable agricultural

decision support systems.

## II. RELATED WORK

### A. Agricultural Decision Support Systems

Early agricultural decision support systems relied on rule-based expert systems incorporating agronomic heuristics and static rule sets. With the advent of machine learning, supervised learning techniques such as Random Forests, Support Vector Machines, and Neural Networks have been widely applied to crop recommendation and yield prediction tasks [8], [9].

However, these systems typically focus on singular predictive objectives and lack integration across advisory, diagnostic, and forecasting domains. Furthermore, most implementations are model-centric rather than architecture-centric.

### B. Vision-Based Crop Disease Detection

Computer vision has been extensively applied to crop disease classification using convolutional neural networks trained on curated leaf datasets [12]. While such approaches demonstrate strong classification performance, they require extensive labeled datasets and retraining for new crop categories

Multimodal vision-language models offer broader generalization capabilities by leveraging pre-trained representations. However, systematic architectural integration of such models within scalable agricultural platforms remains limited.

### C. Time-Series Forecasting in Agricultural Markets

Commodity price forecasting has traditionally employed statistical models such as ARIMA and SARIMA due to their interpretability and computational efficiency [1]. Deep learning approaches including LSTM and Transformer-based forecasting models have been explored but often require large datasets and substantial computational resources.

For infrastructure-constrained environments, lightweight and interpretable models remain attractive for integration within decision support systems.

### D. Hybrid AI Architectures

Hybrid AI architectures combine symbolic reasoning, statistical modeling, and neural inference to leverage complementary strengths [6]. In enterprise systems, microservice-based architectures facilitate modular AI integration through API abstraction layers.

Despite growing adoption in enterprise AI systems, hybrid AI orchestration frameworks tailored for agricultural decision support remain underexplored. This work advances the field by presenting a structured integration strategy designed specifically for agricultural contexts.

## III. PROPOSED HYBRID MULTI-MODAL AI ARCHITECTURE

The proposed framework adopts a layered microservice-oriented architecture to ensure modularity, scalability, and infrastructure efficiency. The architecture is illustrated in Fig. 1.

Fig. 1. Layered Hybrid AI Architecture of the Proposed Framework.

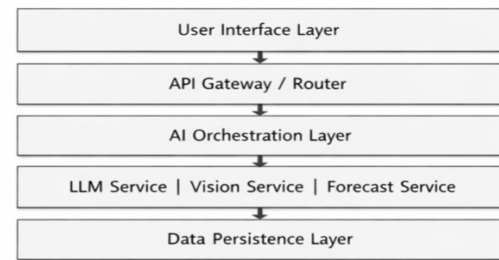


Figure 1. Layered Hybrid AI Architecture of the Proposed Framework.

The architecture consists of five primary layers:

1. Presentation Layer
2. API Gateway Layer
3. AI Orchestration Layer
4. Multi-Modal AI Services Layer
5. Data Persistence Layer

### A. Presentation Layer

The Presentation Layer provides a multilingual user interface

enabling farmers to submit natural language queries, upload crop images, and visualize forecasting outputs. The frontend remains stateless and communicates exclusively through RESTful APIs to ensure separation of concerns.

### B. API Gateway Layer

The API Gateway acts as the centralized routing controller, handling request authentication, validation, and routing. It abstracts internal services from direct exposure and enforces uniform request formatting.

### C. AI Orchestration Layer

The AI Orchestration Layer is the core intelligence mediator of the framework. It performs:

- Request classification
- Context injection
- Model selection
- Structured prompt generation
- API invocation
- Schema enforcement
- Response normalization

This deterministic orchestration pipeline ensures reliable integration of heterogeneous AI modules.

### D. Multi-Modal AI Services Layer

The system integrates three primary AI paradigms:

1. Text-based advisory reasoning via LLM APIs
2. Image-based disease analysis via multimodal vision APIs
3. Statistical price forecasting via ARIMA modelling

Each service is encapsulated within modular abstraction classes to ensure provider independence and fault isolation

### E. Data Persistence Layer

The Data Layer stores user logs, forecasting datasets, and system metadata using relational database management systems. Normalized schema design ensures data integrity and

efficient retrieval.

#### IV. AI ORCHESTRATION PIPELINE

The defining contribution of the proposed framework is its deterministic AI orchestration mechanism, which coordinates heterogeneous AI services within a unified decision-support workflow. The orchestration pipeline is illustrated in Fig. 2.

Fig. 2. Multi-Modal AI Orchestration Pipeline.

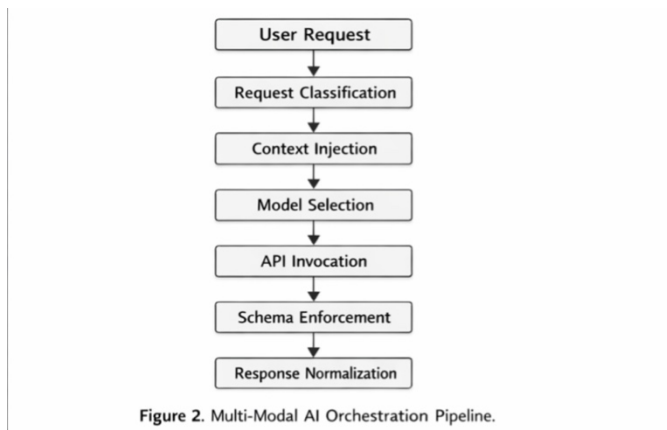


Figure 2. Multi-Modal AI Orchestration Pipeline.

The pipeline consists of the following sequential stages:

1. Request Classification
2. Context Injection
3. Model Selection
4. Structured Prompt Construction
5. API Invocation
6. Schema Enforcement
7. Response Normalization
8. Client Delivery

##### A. Request Classification

Incoming user requests are categorized into one of three primary types:

- Advisory queries (text-based reasoning)
- Disease diagnosis queries (image-based analysis)
- Market forecasting queries (time-series prediction)

Classification is performed using endpoint routing and metadata inspection. This modular routing prevents cross-domain ambiguity.

##### B. Context Injection

Agricultural queries are highly context-sensitive. The orchestration layer augments user inputs with structured contextual parameters including:

- Crop type
- Geographic region
- Season
- Soil characteristics
- Market metadata (if applicable)
- Language preference

This contextual augmentation ensures domain alignment before AI invocation.

##### C. Model Selection and Invocation

Based on classified request type, the orchestration layer selects the corresponding AI module:

- LLM-based advisory engine
- Vision-based diagnostic engine
- ARIMA-based forecasting engine

All AI services are accessed via abstraction wrappers, enabling vendor-neutral integration.

##### D. Schema Enforcement

A critical reliability mechanism is schema-constrained output validation. All LLM-generated outputs must conform to predefined JSON schemas. Invalid responses are either retried or normalized.

This structured validation reduces hallucination risk and improves deterministic behavior.

##### E. Response Normalization

Final responses are standardized into uniform output formats before being delivered to the client interface. This guarantees UI compatibility and consistent user experience.

#### V. MULTI-MODAL INTEGRATION STRATEGY

Unlike isolated agricultural systems, the proposed framework enables cross-modal reasoning across advisory, vision, and forecasting modules.

The integration strategy is illustrated in Fig. 3.

Fig. 3. Hybrid Multi-Modal AI Integration Model.

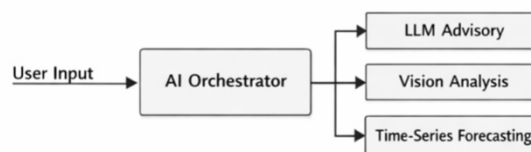


Figure 3. Hybrid Multi-Modal AI Integration Model.

##### A. Advisory–Forecast Coupling

Forecasting outputs influence advisory recommendations. For example:

- Upward price trends may trigger "hold and sell later" guidance.
- Downward trends may prompt immediate selling suggestions.

This coupling enables dynamic economic reasoning.

##### B. Disease–Advisory Coupling

Image-based disease diagnosis is integrated into advisory responses. Once a disease is detected:

- Treatment steps are embedded within advisory content.
- Preventive measures are generated contextually.

This reduces fragmentation between diagnosis and recommendation.

### C. Cross-Module Context Sharing

The orchestration layer allows modules to share contextual metadata. This shared-state mechanism enables composite decision-making rather than isolated inference.

## VI. STRUCTURED PROMPT ENGINEERING FORMALIZATION

Prompt engineering plays a foundational role in ensuring deterministic advisory outputs.

We formalize the prompt construction process as:

$$P = f(U, C, L, S)$$

Where:

- $U$ = User input
- $C$ = Context parameters
- $L$ = Language directive
- $S$ = Output schema constraint
- $f$ = Template transformation function

### A. Role Conditioning

Each advisory prompt begins with domain-specific role definition:

"You are an agricultural advisory expert specializing in regional farming conditions."

Role conditioning restricts generative scope.

### B. Schema-Constrained Output

Outputs are enforced to match structured JSON format:

```
{
  "analysis": "...",
  "recommendations": "...",
  "precautions": "...",
  "language": "...
}
```

Schema validation ensures response reliability and downstream compatibility.

### C. Multilingual Determinism

Language enforcement is embedded directly within prompt construction rather than applied via post-processing translation. This approach improves semantic consistency.

## VII. STATISTICAL FORECASTING INTEGRATION

The forecasting module integrates an Autoregressive Integrated Moving Average (ARIMA) model configured as ARIMA(5,1,0). The forecasting pipeline is illustrated in Fig. 4.

Fig. 4. ARIMA-Based Forecasting Workflow.

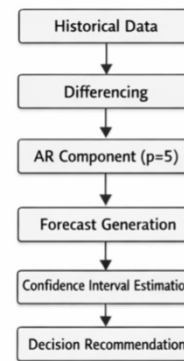


Figure 4. ARIMA-Based Forecasting Workflow.

### A. Mathematical Formulation

The general ARIMA(p, d, q) model is defined as:

$$(1 - \sum_{i=1}^p \phi_i L^i)(1-L)^d y_t = (1 + \sum_{j=1}^q \theta_j L^j) \epsilon_t$$

Where:

- $L$  is the lag operator
- $\phi_i$  are autoregressive coefficients
- $\theta_j$  are moving average coefficients
- $d$  is differencing order
- $\epsilon_t$  is white noise

For the implemented configuration:

$$p = 5, d = 1, q = 0$$

This yields:

$$(1 - \sum \phi_i L^i)(1 - L)^d y_t = (1 + \sum \theta_i L^i) \epsilon_t$$

### B. Stationarity and Differencing

First-order differencing ensures stationarity:

$$\Delta y_t = y_t - y_{t-1}$$

This stabilizes the mean and enables autoregressive modeling.

### C. Forecast Utilization

Forecast outputs include:

- Short-term price prediction
- Confidence intervals
- Trend direction classification

These outputs are injected into advisory modules for integrated decision support.

Statistical forecasting approaches such as ARIMA remain particularly suitable for agricultural decision support due to their interpretability and minimal computational requirements. Unlike deep learning forecasting models that require extensive training datasets and specialized hardware accelerators, ARIMA models can operate efficiently using limited historical commodity price data. This property makes ARIMA-based forecasting practical for deployment in rural agricultural environments where computational infrastructure may be constrained. Additionally, the transparency of autoregressive models allows agricultural advisors and policymakers to interpret predicted price trends and confidence intervals, improving trust in automated market-aware decision support recommendations.

## VIII. EXPERIMENTAL EVALUATION

The evaluation of the proposed framework focuses on architectural robustness, functional reliability, and modular integration stability rather than large-scale predictive benchmarking. The goal is to validate the feasibility of hybrid multi-modal AI orchestration in an agricultural decision-support context.

### A. Evaluation Objectives

The experimental validation addresses the following objectives:

1. Functional correctness of the orchestration pipeline
2. Structured schema compliance of AI outputs
3. Stability of multi-modal routing
4. Forecasting pipeline integrity
5. Observational latency characteristics

### B. Functional Validation

#### 1) Advisory Module Testing

Text-based advisory queries were tested across multiple agricultural scenarios including:

- Crop planning recommendations
- Seasonal farming advisory
- Market timing suggestions

Each response was evaluated for:

- JSON schema compliance
- Context inclusion correctness
- Multilingual formatting consistency
- Logical coherence of recommendations

All valid responses adhered to predefined schema constraints. Beyond functional correctness, the modular orchestration architecture was also evaluated for integration stability across heterogeneous AI services. The orchestration layer maintained consistent response formatting when advisory reasoning, disease diagnosis, and forecasting modules were invoked simultaneously. This demonstrates that the structured orchestration pipeline successfully coordinates multiple AI services without cross-module interference. The abstraction-based service design further enables individual AI components to be updated or replaced without affecting the overall system workflow, improving maintainability and long-term system robustness.

#### 2) Vision-Based Disease Analysis Testing

Image-based diagnosis was evaluated using sample crop leaf images representing common disease categories.

Validation criteria included:

- Proper image encoding and API invocation
- Disease identification output structure
- Integration of treatment recommendations
- Advisory coupling consistency

The orchestration layer successfully routed diagnostic outputs into advisory modules for integrated responses.

#### 3) Forecasting Pipeline Validation

The ARIMA(5,1,0) model was validated using statistically structured synthetic commodity price sequences to demonstrate forecasting pipeline functionality.

Evaluation included:

- Stationarity verification after differencing

- Autoregressive coefficient stability
- Forecast generation consistency
- Confidence interval estimation

The forecasting outputs were successfully integrated into advisory reasoning for market-aware recommendations.

### C. Observational Latency Analysis

Although no dedicated benchmarking framework was deployed, qualitative latency observations indicated:

- Text-based advisory latency primarily dependent on external LLM inference time
- Vision analysis latency dependent on image encoding size and API processing time
- ARIMA forecasting latency negligible due to local computation

Future quantitative evaluation will incorporate timestamp-based logging and statistical latency measurement.

### D. Proposed Quantitative Metrics for Future Work

For empirical benchmarking in future deployment phases, the following metrics are proposed:

- Root Mean Squared Error (RMSE) for forecasting
- Mean Absolute Error (MAE) for price prediction
- Schema Compliance Rate for advisory responses
- Average API Response Time
- Throughput under concurrent request simulation

These metrics establish a foundation for future large-scale evaluation.

## IX. SCALABILITY AND DEPLOYMENT ANALYSIS

Scalability is a primary architectural objective of the proposed framework.

### A. Microservice-Based Modularity

The framework follows a microservice-oriented design where:

- Advisory service
- Vision service
- Forecasting service

are implemented as independent modules accessed via RESTful APIs.

This modular design enables:

- Independent scaling of services
- Fault isolation
- Provider replacement without architectural redesign

The microservice-based architecture also supports horizontal scalability through independent service deployment. Individual AI services such as advisory reasoning, image-based disease detection, and forecasting analysis can be scaled independently depending on demand patterns. For instance, image analysis workloads may increase during peak crop disease monitoring periods, while advisory reasoning services may experience continuous usage throughout the agricultural cycle. By separating these

services through API-based orchestration, the framework enables flexible infrastructure scaling while maintaining stable system performance.

### B. Stateless Frontend and API Gateway

The presentation layer remains stateless, enabling horizontal scaling via load balancing mechanisms. The API gateway enforces:

- Authentication
- Request validation
- Uniform routing

This separation enhances maintainability and extensibility.

### C. Infrastructure Efficiency

Unlike monolithic deep learning deployments requiring GPU infrastructure, the proposed architecture:

- Leverages external AI APIs for inference
- Performs lightweight statistical forecasting locally
- Minimizes hardware requirements

This makes the framework suitable for deployment in infrastructure-constrained rural environments.

## X. LIMITATIONS

Despite its architectural strengths, the framework has several limitations.

### A. API Dependency

Reliance on external AI providers introduces potential variability in response behavior and latency.

### B. Synthetic Forecasting Data

The current validation uses statistically structured synthetic datasets rather than live agricultural market data. Future integration with real-time agricultural market APIs is required for empirical validation.

### C. Absence of Field Trials

Large-scale deployment across rural farming communities has not yet been conducted. Real-world usability studies remain future work.

## XI. CONCLUSION

This paper presented a hybrid multi-modal AI orchestration framework designed to support practical agricultural decision-making. Unlike traditional approaches that focus on individual model performance, the proposed system emphasizes the integration of multiple AI components into a unified workflow.

The framework combines language-based advisory reasoning, image-based disease diagnosis, and ARIMA-based price forecasting within a structured orchestration pipeline. By introducing stages such as request classification, context injection, and schema-based response validation, the system ensures consistent and reliable outputs across different types of queries.

The microservice-based architecture further enhances modularity and scalability, allowing individual components to be updated or replaced without affecting the overall system. This design makes

the framework suitable for deployment in real-world agricultural environments where flexibility and resource efficiency are important.

While the current implementation demonstrates the feasibility of multi-modal AI orchestration, future work will focus on integrating real-time agricultural datasets, conducting large-scale field evaluations, and improving quantitative performance benchmarking.

Overall, this work highlights the importance of system-level design in developing next-generation agricultural decision support platforms that go beyond isolated machine learning models.

## REFERENCES

- [1] G. E. Box and G. M. Jenkins, *Time Series Analysis: Forecasting and Control*. San Francisco, CA, USA: Holden-Day, 1976.
- [2] J. Brownlee, *Introduction to Time Series Forecasting with Python*. Melbourne, Australia: Machine Learning Mastery, 2018.
- [3] T. Wolf et al., "Transformers: State-of-the-Art Natural Language Processing," in *Proc. Conf. Empirical Methods Natural Language Processing (EMNLP)*, 2020.
- [4] J. Devlin, M. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," in *Proc. NAACL-HLT*, 2019.
- [5] A. Krizhevsky, I. Sutskever, and G. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," in *Proc. Advances in Neural Information Processing Systems*, 2012.
- [6] S. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach*, 3rd ed. Pearson Education, 2016.
- [7] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.
- [8] A. Kamilaris and F. X. Prenafeta-Boldú, "Deep Learning in Agriculture: A Survey," *Computers and Electronics in Agriculture*, vol. 147, pp. 70–90, 2018.
- [9] K. Liakos et al., "Machine Learning in Agriculture: A Review," *Sensors*, vol. 18, no. 8, 2018.
- [10] J. Fountas et al., "Farm Management Information Systems: Current Situation and Future Perspectives," *Computers and Electronics in Agriculture*, vol. 115, pp. 40–50, 2015.
- [11] M. Jones et al., "Artificial Intelligence Applications in Agriculture," *Agricultural Systems*, vol. 187, pp. 102994, 2020.
- [12] H. Zhang et al., "Deep Learning-Based Crop Disease Detection: A Review," *Computers and Electronics in Agriculture*, vol. 180, 2021.