AgriVerify: An AI-Based Verification System for Agricultural LLM Responses

T Somasekhar
Associate Professor
Department of Computer Science and Engineering
K S Institute of Technology
Bengaluru, Karnataka, India
somasekhart@ksit.edu.in

Lavanya R
Student
Department of Computer Science and Engineering
K S Institute of Technology
Bengaluru, Karnataka, India
lavanyarameshvc@gmail.com

Pavan Gowda S
Student
Department of Computer Science and Engineering
K S Institute of Technology
Bengaluru, Karnataka, India
pavangowda30911@gmail.com

***Abstract*—** Large language models (LLMs) can answer agricultural questions fluently, but may include unsupported or incorrect claims. AgriVerify checks generated answers by extracting factual claims and retrieving evidence from a local agricultural dataset and online sources. Semantic similarity identifies relevant passages, while natural-language inference estimates support, contradiction, or neutrality. Source credibility, freshness, agreement, and hallucination risk contribute to claim-level confidence and verdicts. The system labels claims as supported, contradicted, or uncertain, then uses verified evidence to prepare a corrected answer. Implemented in Streamlit, it runs llama3.2:3b locally through Ollama and provides evidence, scores, results, and PDF reports. The local collection has 315 records in 22 categories, with no missing fields or duplicate questions. An archived ten-claim regression test reached 90% accuracy (F1 = 0.923), but larger expert-labeled agricultural evaluation remains necessary.

Keywords— agriculture, fact verification, hallucination detection, large language models, retrieval-augmented generation.

I. INTRODUCTION

Farmers, students, and agricultural professionals often need short answers about crops, soil, irrigation, nutrients, pests, and diseases. LLMs can provide these answers quickly and have performed well on agricultural question-answering tasks [1, 2]. However, a fluent answer may still contain unsupported facts, omit an important condition, or use outdated information. This is important in agriculture because advice may change with crop variety, location, weather, soil, and growth stage.

Retrieval-augmented generation (RAG) gives external information to a model while it prepares an answer [3]. Agricultural retrieval systems have also used embeddings and fine-grained semantic search [4, 5]. Retrieval alone does not verify every generated statement, since the selected passage may be partly relevant and the model may add new details. AgriVerify therefore performs a separate check after generation.

The application runs Llama 3.2 3B locally through Ollama, retrieves local and online evidence, extracts factual claims, and verifies them separately. Semantic similarity, natural-language inference (NLI), credibility, freshness, source agreement, contradiction, and hallucination risk are shown in the Streamlit interface and used to prepare a corrected answer.

The main contributions are:

- a local, end-to-end agricultural response-verification workflow;
- claim-level checking using local and online evidence;
- visible evidence, scores, verdicts, and corrected responses; and
- an implementation audit and preliminary regression check.

II. RELATED WORK

Agricultural LLM studies have examined model accuracy, prompting, RAG, and multilingual data. Silva et al. evaluated several general-purpose LLMs on agricultural questions, while Eldem and Eldem compared prompting methods for agricultural question answering [1, 2]. Kaur et al. showed the value of multilingual, domain-specific evaluation [6]. These studies assess the final answer but do not check every factual statement in free text.

Agricultural retrieval work includes FINDER and Sem-RAG, which use embeddings and fine-grained semantic stores to improve access to domain knowledge [4, 5]. In general fact verification, FEVER and FEVEROUS organize the task around claim identification, evidence retrieval, and verdict prediction [7, 8]. Recent systems commonly combine retrieval, NLI, prompting, and explanation generation [9].

A. Research Gap

Few agricultural systems combine local answer generation, claim extraction, claim-specific evidence, visible verification scores, and answer correction in one application. AgriVerify addresses this gap as an implementation-focused prototype.

III. PROPOSED SYSTEM

A. System Objective

AgriVerify checks an answer one claim at a time rather than treating the whole paragraph as verified. Questions outside agriculture are rejected early. Accepted questions use local and online evidence for generation and verification. Weak evidence produces an uncertain verdict rather than a forced decision. Model requests run locally through Ollama, while the interface shows the evidence and scores used for each result.

B. Overall System Workflow

Figure 1 shows the workflow. The application checks the question, retrieves local and online evidence, and asks Llama to prepare an initial answer. It then extracts up to six factual

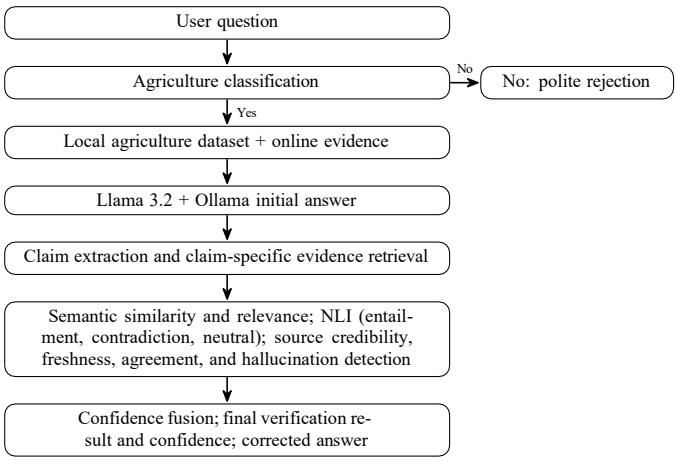


Figure 1. Workflow of the proposed agricultural response-verification system.

claims. Each claim starts a separate evidence search and is checked using relevance, NLI, credibility, freshness, agreement, and hallucination risk. The final stage assigns a verdict, displays the evidence and confidence, and prepares a corrected answer.

C. Major System Modules

1) Agriculture Domain Classification

The domain gate checks whether the question belongs to agriculture before retrieval or answer generation begins. A normalized keyword check handles clear questions about crops, soil, irrigation, nutrients, pests, diseases, livestock, and farm management. When the keyword result is weak, Llama performs a second classification. Questions outside the domain are stopped early so that the application does not present unrelated answers as agricultural guidance.

2) Local and Online Evidence Retrieval

The evidence module first compares the question with the local CSV collection and retains the three highest nonzero matches. It then prepares agriculture-focused online searches and keeps up to six accepted sources. Government, university, FAO, USDA, ICAR, and recognized scientific or agricultural domains receive preference. Duplicate URLs and sources below the credibility threshold are removed before the evidence is used by the model.

3) Answer Generation and Claim Extraction

The generation module combines the retrieved local and online material in the first prompt. Llama is asked to answer only the agricultural question, avoid unsupported quantities or dates, and state uncertainty when the evidence is incomplete. A second prompt separates the answer into no more than six factual claims in JSON format. If the returned JSON cannot be parsed, the application uses sufficiently long sentences as fallback claims.

4) Claim Verification

Each extracted claim begins a separate verification cycle. New online evidence is retrieved for the claim, and the best passage is selected using semantic similarity and source credibility. Llama then estimates entailment, contradiction, and neutral values. The system also calculates freshness, agreement between sources, and hallucination risk. Keeping these values separately allows the interface to show why a claim received its result.

5) Confidence and Verdict Generation

The confidence module combines relevance, entailment, credibility, freshness, and agreement, while subtracting hallucination and contradiction penalties. A strong contradiction produces a contradicted label. A claim is supported only when the relevance, entailment, and risk conditions all pass their thresholds. Every other case is marked uncertain, which prevents weak or missing evidence from being treated as proof.

6) Correction and Report Generation

The correction module receives the first answer, claim verdicts, and selected evidence. Supported details are retained, contradicted details are corrected or removed, and uncertain details are qualified. The same records are used to prepare the downloadable PDF report. This report includes the question, first and corrected answers, claim labels, confidence values, verification signals, and evidence sources.

7) Result Visualization

The Streamlit interface presents the workflow in the same order in which it is executed. Metric cards show the overall status and claim counts, while progress bars and claim panels show confidence and verification signals. Expandable sections allow the user to inspect the best evidence and remaining sources. This design keeps the verification process visible instead of returning only a final percentage.

8) Failure Handling and Conservative Behaviour

The application handles model, search, and parsing failures without treating them as successful verification. Invalid JSON from the model activates a sentence-based claim fallback, while invalid NLI output is treated as neutral. When online evidence is unavailable, hallucination risk is increased and the claim normally becomes uncertain. Connection errors and timeouts are shown in the interface so that the user can distinguish a system failure from a factual verdict.

IV. ARCHITECTURE AND VERIFICATION METHOD

Python manages retrieval, model requests, scoring, and reports; Streamlit presents the results. We run llama3.2:3b locally through Ollama [10]. Local records and online results guide answer generation, while new online evidence is retrieved to check each claim.

A. Domain Classification and Evidence Retrieval

We normalize the question and check agricultural keywords. When keywords match, the domain score is 0.55 plus 0.10 per distinct match, capped at 0.95. A positive score of at least 0.65 is accepted; otherwise, Llama makes the domain decision.

For local retrieval, we divide the number of shared distinct words by the number of distinct query words. Each record's question, category, answer, and keywords are searched. Empty queries return no matches; the three highest nonzero matches are retained. Online results are filtered using domain-based

credibility rules.

B. Answer Generation and Claim Extraction

The answer prompt includes the retrieved material and uses temperature 0.2 with a 700-token limit. A second prompt requests up to six factual claims in JSON. If parsing fails, sufficiently long sentences become fallback claims.

C. Claim-Level Verification

Each claim is compared with the titles and snippets returned by the evidence search. The application normally uses all-MiniLM-L6-v2 similarity [11]; if that model is unavailable, it falls back to lexical Jaccard similarity. Evidence relevance gives 80% weight to similarity and 20% to source credibility. This prevents a trusted but unrelated page from being selected only because of its domain. The source with the highest score becomes the main evidence for the claim.

Llama then compares the selected evidence with the claim and returns support, contradiction, and neutral scores. Hallucination risk increases when relevance or support is weak and when contradiction or neutrality is high. If no evidence is found, the risk is set to 0.90. Values below 0.40 are treated as low risk, values from 0.40 to 0.69 as medium risk, and values from 0.70 as high risk.

Source agreement is estimated from similarity across the retrieved results. A single result uses its similarity score. For several results, the application combines the proportion with similarity of at least 0.35 and the average similarity. Freshness is estimated from the newest year found in the selected snippet. When no year is available, the system assigns a cautious value based on source credibility. This value is only an estimate because a year in a snippet may not be the publication date.

D. Confidence Fusion and Verdict Generation

Table I summarizes the values used by the current implementation. Relevance, entailment, credibility, freshness, and source agreement increase confidence. Hallucination risk and contradiction reduce it. The final value is limited to the range from 0 to 1. These weights were selected manually for the prototype and should not be read as calibrated probabilities.

The verdict is decided separately from the confidence percentage. Contradiction is checked first. A claim is supported only when relevance and entailment pass their thresholds and hallucination risk remains below its limit. All remaining claims are marked uncertain, so missing evidence is not treated as proof that a statement is false.

Any contradicted claim makes the answer partially contradicted. If all claims are supported, the answer is supported; if only some are supported, it is partially verified. Otherwise, it is uncertain. Overall confidence is the mean claim score, shown alongside the verdicts and evidence.

V. IMPLEMENTATION

A. Streamlit User Interface

The integrated Streamlit application follows the workflow in Fig. 1. It shows the domain decision, retrieved evidence, generated answer, extracted claims, claim cards, overall status, corrected answer, charts, and report download. Figure 2 presents the complete drip-irrigation demonstration in the same order as the system workflow.

TABLE I
VERIFICATION SETTINGS USED IN THE PROTOTYPE

Setting	Implemented value
Evidence relevance	Similarity 80%; credibility 20%
Hallucination risk	Weak relevance 40%; weak entailment 35%; contradiction 15%; neutral 10%
Confidence gains	Relevance 25%; entailment 30%; credibility 15%; freshness 10%; agreement 20%
Confidence penalties	Hallucination 20%; contradiction 25%
Contradicted verdict	Contradiction score at least 0.65
Supported verdict	Relevance at least 0.45, entailment at least 0.55, and risk below 0.60
Other cases	Uncertain

TABLE II
PROFILE OF THE LOCAL AGRICULTURAL DATASET

Property	Value
Question–answer records	315
Agricultural categories	22
Missing fields	0
Duplicate questions	0
Mean question length	6.33 words
Mean answer length	14.16 words

B. Visualization Dashboard

The dashboard uses metric cards, progress bars, charts, status messages, and source expanders. The displayed values come from the same claim records used for the final verdict.

C. Ollama and Llama Integration

The application calls Ollama through /api/generate and checks /api/tags for the required model. Classification, claim extraction, NLI, and correction use temperature 0; initial answer generation uses 0.2. Errors and timeouts are reported in the interface.

D. Runtime and Error Handling

Before processing a question, the application checks whether the Ollama service is reachable and whether the selected model is installed. Model calls use bounded output lengths and a request timeout so that one stalled stage does not block the complete interface. Search failures, empty evidence lists, invalid model responses, and report-generation errors are converted into readable status messages. These checks are important because a verification system should not display a confident result when an internal stage did not complete.

E. Local Agricultural Dataset

The CSV contains questions, categories, answers, and keywords. Table II summarizes the audit. The four largest categories contain 76.2% of the records, so the collection is useful but not balanced.

F. Evidence and Semantic Processing

DDGS supplies online titles and snippets, with preference given to government, university, FAO, USDA, and ICAR material. Sentence Transformers supplies cosine similarity when available; otherwise, the system falls back to lexical similarity.

G. Source Handling and Report Generation

For every accepted online result, the application keeps the title, snippet, URL, search query, and credibility value. Duplicate links are removed before the evidence is passed to the model. This prevents the same page from being counted several times during agreement calculation. Local records and online sources are kept separately so that the interface can show where the supporting information came from.

After all claims are checked, the correction prompt receives the original question, first answer, claim verdicts, and selected evidence. It is instructed to retain supported statements, correct contradicted statements, qualify uncertain details, and avoid adding new facts. The PDF report records the question, first answer, corrected answer, claim verdicts, confidence values, evidence signals, and source details. The report is intended to preserve the reasoning path shown in the Streamlit interface.

VI. RESULTS AND DISCUSSION

A. Prototype Demonstration

The application completed the full path from question entry to corrected answer, as shown in Fig. 2. In the drip-irrigation demonstration, three claims were extracted and all were marked supported, with an overall confidence of 81.9%. The user could inspect the evidence and identify whether a lower score came from relevance, contradiction, freshness, credibility, or agreement. This run demonstrates the interface and is not a benchmark result. The dataset audit also confirmed that all 315 records contain the required fields and that no questions are duplicated.

B. Functional Observations

The order of the interface follows the verification process rather than showing only a final answer. The user first sees whether the question belongs to agriculture and which evidence was retrieved. The generated answer and extracted claims are then shown before the verdicts. This arrangement makes it easier to notice whether an error came from weak evidence, an uncertain NLI result, an old source, or disagreement between sources.

The supported, contradicted, and uncertain labels also have different meanings. Supported means that relevant evidence and entailment passed the configured thresholds. Contradicted means that the contradiction score was high. Uncertain means that the available evidence was not strong enough for either decision. Therefore, a missing source is not automatically treated as proof that a claim is false.

C. Dataset Audit Results

The local collection passed the basic structural checks. Every record contains a question, category, answer, and keyword value, and no question is repeated. The short answers are useful as retrieval context because they do not add long unrelated passages to the prompt. However, 76.2% of the records belong to four categories. This imbalance means that the local collection is more useful for common crop questions than for livestock, postharvest management, farm economics, and regional recommendations.

TABLE III
ARCHIVED REGRESSION TEST ($n = 10$)

Method	Acc.	Prec.	Rec.	F1	Brier
NLI baseline	0.900	0.857	1.000	0.923	0.070
Three-signal fusion	0.900	0.857	1.000	0.923	0.140

D. Archived Pilot Regression Test

The archive contains ten labeled general-knowledge claims from an earlier offline branch: six supported and four refuted. Because that branch differs from the current application, its stored predictions are used only as a regression check. At a threshold of 0.5, both reported methods produce six true positives, three true negatives, one false positive, and no false negatives.

The false positive states that the Sun revolves around the Earth. The fusion method does not correct it and has a poorer Brier score than the NLI baseline. No calibrated result is reported because a separate calibration set is unavailable.

Accuracy records the fraction of correct labels, while precision and recall describe the supported class. The F1-score combines precision and recall. The Brier score measures the difference between predicted confidence and the observed label, so a smaller value is better. In this small test, the fusion method gives the same class predictions as the NLI baseline but produces a poorer Brier score. This is one reason the current paper does not claim that confidence fusion is already calibrated.

E. Discussion

AgriVerify makes claim-level evidence and scores visible and avoids a paid model API. Its main tradeoff is independence: the same Llama model handles generation, classification, claim extraction, NLI, and correction, so related errors are possible. The system is also hybrid rather than fully offline because claim verification depends on online snippets.

The confidence value should be read together with the claim verdict and evidence. A high percentage does not by itself prove that agricultural advice is suitable for every farm. The crop, location, soil, weather, growth stage, and date of the source still matter. For this reason, the interface keeps the evidence visible and produces an uncertain result when the retrieved material is weak.

Local model execution keeps the prompts and generated text on the host computer and avoids a paid model service. However, the application is not completely offline because current claim verification depends on online evidence. If search is unavailable, the system increases hallucination risk and normally returns an uncertain result. This behavior is safer than treating the model's stored knowledge as verified evidence.

F. Practical Deployment Considerations

The interface is intended to support review rather than hide the verification process. A user should be able to read the corrected answer, open the evidence list, and understand why a claim received its label. The PDF report provides a record of the same values, but it should not be treated as a certificate that the answer is universally correct. Agricultural recommendations can change across districts, seasons, crop varieties, and farm conditions, even when the general statement is supported.

The local model setup also affects practical use. Ollama and the selected model must be installed on the host computer, and response time depends on the available processor, memory, and number of claims. Each additional claim can start another evidence search and NLI request. A production version should therefore measure end-to-end latency, model failures, search failures, and report-generation time on the hardware used by the target users.

Questions and model prompts are processed locally, but online retrieval sends search terms to an external search service and the report may contain source URLs and generated text. A deployment should therefore explain which information remains local and which information is sent outside the computer. Saved reports should be stored securely when questions contain farm-specific, location-specific, or commercially sensitive details.

G. Responsible Use

Recommendations involving pesticide use, fertilizer quantities, animal health, or regulated products require additional caution because general evidence may not match the user's crop, location, weather, or field condition. When sources disagree or local confirmation is required, the corrected answer should show the uncertainty and direct the user to an official extension source or qualified agricultural professional.

VII. LIMITATIONS

The evaluation contains only ten general-domain claims and cannot measure performance across the 22 agricultural categories. A larger set of agricultural claims should be labeled by domain experts and divided into separate development and test sets. The current accuracy and F1 values are therefore reported only as a regression check.

Online verification uses search snippets rather than saved full passages. A snippet may leave out conditions, units, or regional details that are present on the complete page. Credibility is also assigned using domain rules, and the confidence weights are manually selected rather than learned from agricultural verification data.

Freshness is estimated from years found in the retrieved text. A year mentioned in a snippet is not always the publication date, so this signal can be inaccurate. A later version should download the complete passage, read verified page metadata, validate the host name strictly, and preserve the evidence used for each report.

The same Llama model is used at several stages, which may produce related errors. Location, crop variety, soil tests, weather, growth stage, registered products, and local regulations are not represented systematically. The output should therefore support decisions, not replace an agronomist or official extension advice.

TABLE IV
PLANNED EVALUATION MEASURES

Stage	Measure
Domain gate	Accuracy and macro-F1
Claim extraction	Expert precision and recall
Evidence retrieval	Recall@k and source relevance
Claim verdict	Macro-F1 and confusion matrix
Confidence	Brier score and calibration error
Complete system	Latency and user usefulness ratings

The current online results are not stored as a fixed evidence collection. Search rankings and snippets may change over time, which makes an earlier verification result difficult to reproduce exactly. Future versions should save the selected passage, retrieval date, page title, host name, and verified publication date together with the report. This would allow a later reviewer to see the same material that was used for the verdict.

The present evaluation does not measure retrieval recall. A correct claim may be marked uncertain because the search process failed to find the right source, while a relevant-looking snippet may still lack the context required for a sound decision. A future benchmark should therefore evaluate domain classification, claim extraction, evidence retrieval, verdict prediction, confidence calibration, and corrected-answer quality as separate stages.

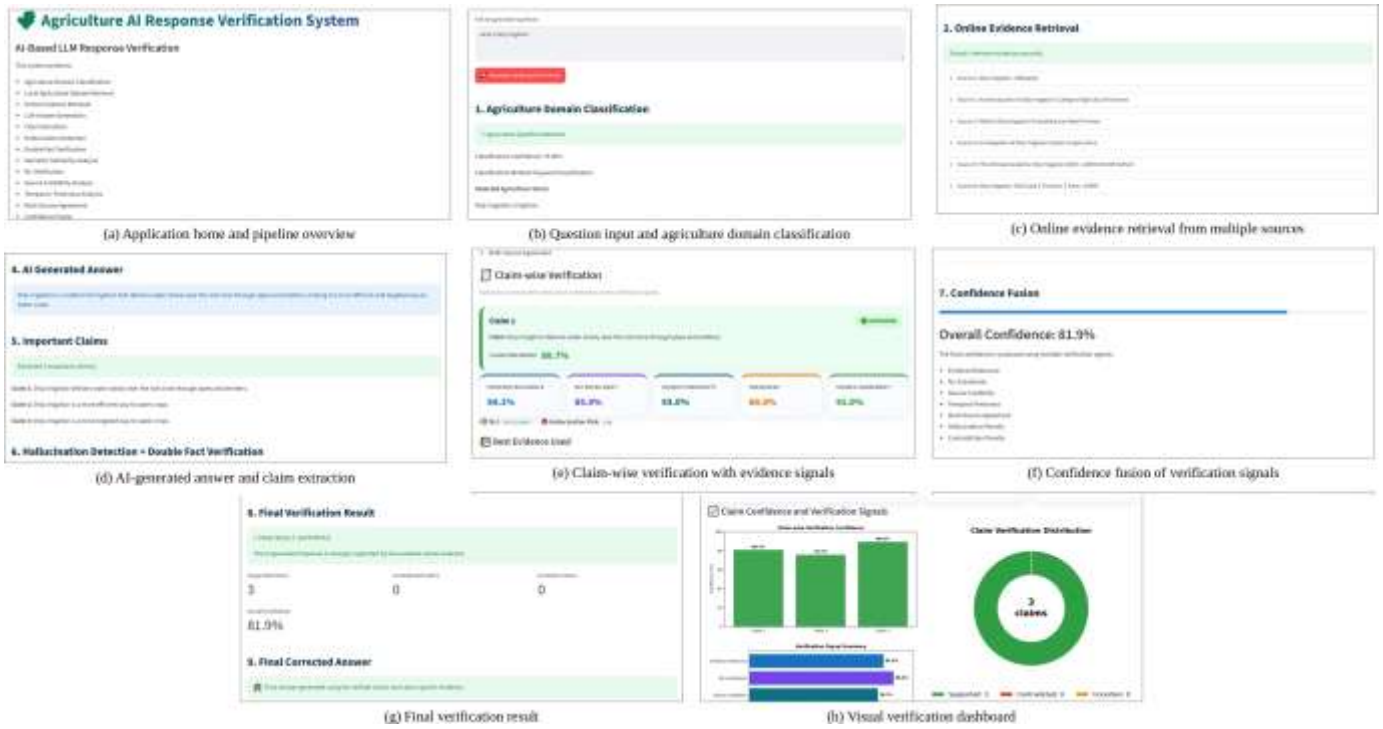
The Streamlit demonstration shows one supported response, but it does not establish performance for contradicted and uncertain cases. Further testing should include incorrect quantities, reversed relationships, outdated recommendations, location-dependent advice, and questions with no reliable evidence. Review by agricultural experts is also needed to determine whether the explanations and caution messages are useful in practice.

A. Planned Evaluation

A complete evaluation should test each stage separately before reporting one overall system score. Domain classification should be tested with both agricultural and unrelated questions. Claim extraction should be compared with claims identified by human reviewers. Evidence retrieval should be checked against sources selected by agricultural experts, while verdict prediction should include balanced supported, contradicted, and uncertain examples.

Table IV lists the measures planned for the next evaluation. Reporting the stages separately will show whether an incorrect final result began with classification, claim extraction, retrieval, NLI, or confidence fusion. The test data should be kept separate from the examples used to select thresholds and confidence weights.

User studies should include participants with different levels of agricultural knowledge. They can assess whether the evidence, uncertainty messages, corrected answer, and downloadable report are understandable and useful.



(a) Application home and pipeline overview (b) Question input and agriculture domain classification (c) Online evidence retrieval from multiple sources

(d) AI-generated answer and claim extraction (e) Claim-wise verification with evidence signals (f) Confidence fusion of verification signals

(g) Final verification result (h) Visual verification dashboard

Figure 2. Workflow-ordered AgriVerify interface demonstration: (a) application overview, (b) question input and domain classification, (c) online evidence retrieval, (d) answer and claim extraction, (e) claim-wise verification, (f) confidence fusion, (g) final result, and (h) verification dashboard.

VIII. CONCLUSION

AgriVerify combines agricultural answer generation with claim-level evidence checking. It shows the evidence and verification signals, assigns a verdict to each claim, prepares a corrected answer, and creates a PDF report. The prototype completes the intended workflow, but the archived ten-claim test is too small to represent agricultural performance. Future work will use expert-labeled agricultural claims, complete evidence passages, verified source dates, an independent verifier, calibrated confidence, and user testing before the system is considered ready for practical decision support.

REFERENCES

[1] B. Silva, L. Nunes, R. Estevão, V. Aski, and R. Chandra, "GPT-4 as an agronomist assistant? Answering agriculture questions using large language models," arXiv preprint arXiv:2310.06225, 2023.

[2] A. Eldem and H. Eldem, "The development and evaluation of agricultural question-answering systems based on large language models," *Scientific Reports*, vol. 16, p. 5357, 2026.

[3] P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, H. Küttler, M. Lewis, W. tau Yih, T. Rocktäschel, S. Riedel, and D. Kiela, "Retrieval-augmented generation for knowledge-intensive NLP tasks," in *Advances in Neural Information Processing Systems*, vol. 33, 2020, pp. 9459–9474.

[4] R. Peng, K. Liu, P. Yang, Z. Yuan, and S. Li, "Embedding-based retrieval with LLM for effective agriculture information extracting from unstructured data," arXiv preprint arXiv:2308.03107, 2023.

[5] B. Bai, X. Meng, and C. Zhao, "Research on Sem-RAG: A corn planting knowledge question-answering algorithm based on fine-grained semantic information retrieval enhancement," *Applied Sciences*, vol. 15, no. 19, p. 10850, 2025.

[6] R. Kaur, A. S. Bhankhar, J. S. Salh, S. Rajput, Vidhi, K. Mahendra, B. Berwal, R. Kumar, and S. Ranathunga, "Leveraging synthetic data for question answering with multilingual LLMs in the agricultural domain," arXiv preprint arXiv:2507.16974, 2026.

[7] J. Thorne, A. Vlachos, C. Christodoulopoulos, and A. Mittal, "FEVER: A large-scale dataset for fact extraction and verification," in *Proceedings of NAACL-HLT*, 2018, pp. 809–819.

[8] O. Temiz, Ö. O. Kılıç, A. O. Kızıldağ, and T. T. Temizel, "A fact checking and verification system for FEVEROUS using a zero-shot learning approach," in *Proceedings of the Fourth Workshop on Fact Extraction and Verification*, 2021, pp. 113–120.

[9] A. Dmonte, R. Oruche, M. Zampieri, P. Calyam, and I. Augenstein, "Claim verification in the age of large language models: A survey," in *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics, Student Research Workshop*, 2026, pp. 15–29.

[10] A. Dubey et al., "The Llama 3 herd of models," arXiv preprint arXiv:2407.21783, 2024.

[11] N. Reimers and I. Gurevych, "Sentence-BERT: Sentence embeddings using siamese BERT-networks," in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*, 2019, pp. 3982–3992.