International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 10 Issue: 10 | Oct - 2026 SJIF Rating: 8.659 ISSN: 2582-3930

VeriAd: A CRAG-Orchestrated Multimodal Compliance Auditing Framework for Short-Form Advertisement Videos with Temporal Evidence Grounding and Cross-Modal Contradiction Detection

Ashish P Mohod [1], Ayush P Kurzekar [2], Harsh G Sewalkar [3], Lokesh U Parse [4], Mohammed Aasim [5], Shreyash S Malvi [6]

[2, 3,4,5,6] Student, Department of Computer Science and Engineering - II,
[Priyadarshini Bhagwati college of Engineering], [Nagpur Maharashtra], India
[1] Assistant Professor, Department of Computer Science and Engineering - II,
[Priyadarshini Bhagwati college of Engineering], [Nagpur Maharashtra], India

---

Abstract - Short form video ads are created and posted at a much higher rate than current compliance teams are able to process. Automated systems will typically only run checks using the audio transcription or the retrieved text as a whole and therefore fail to detect when the voice over and text do not match. This paper details VeriAd which is a framework built on Azure for auditing video advertisements based on advertising codes and platform policies. A video is uploaded and stored within Azure Blob Storage. Azure AI Video Indexer indexes the video to give a time stamped transcription of the audio, text on screen and visual tags. This data is combined into a single Temporal Evidence Timeline. The claims and disclaimers are extracted from this. A Corrective Retrieval Augmented Generation loop is used to retrieve and score the rules based on the jurisdiction. This uses Azure AI Search to retrieve the relevant clauses of the rules. The cross-modal contradiction module scores the disclaimers to ensure that they are similar enough, long enough, legible and consistent enough to pair with the claim. The workflow is stateful and is orchestrated using LangGraph. Azure OpenAI GPT-4o provides evidence-based reasoning. All steps are logged using LangSmith. The result is a pass/fail/needs review decision that includes times, rules and suggested resolutions. A methodology is also provided to evaluate the system using 5 different ablation scenarios.

Keywords: Advertisement Compliance, Multimodal Video Analysis, Corrective Retrieval-Augmented Generation, LangGraph, Temporal Evidence Grounding, Cross-Modal Contradiction Detection, Human-in-the-Loop Review

1. INTRODUCTION

Much of digital advertising is now in the form of 15 to 60 second video clips. These may present a voiceover, quick text display, product shot, price list and small disclaimer in just a few seconds. Advertisements must be free from misleading information. This includes misleading through implication, omission, ambiguity and exaggeration. All factual statements must be capable of supporting evidence [13]. Additionally, the disclaimer must not attempt to counter any misleading claims and should be consistent with any claims made [21]. In the US, the FTC requires qualifying information to be prominent and conspicuous. They have officially defined this phrase in their 2023 updated Endorsement Guides [20, 25].

Manual verification of these conditions can be time consuming and inconsistent. Each video must be viewed, the text overlays paused and the appropriate clause looked up. However, larger platforms are now using Machine Learning techniques to help with this task. For one policy of Google Ads, LLM-based verification and clustering reduced the number of required verifications by over three orders of magnitude [17]. Research has also been done very quickly. RAVEN and RAVEN++ use Reinforcement Learning to train a Multimodal Language Model to detect and locate claims in advertising videos [10, 11]. BLM-Guard uses models to detect cross modal inconsistencies such as subtitle and speech drift [2].

However, most of these are trained models requiring large internal datasets. These would be impossible to replicate by a college team, small agency or individual. There are lighter Retrieval Augmented Generation (RAG) [24] systems however they often believe everything the retrieve provides. This can cause inaccurate outputs if the retrieved rule is incorrect or not very relevant. Rarely will these systems verify if the disclaimer actually qualifies the claim.

VeriAd is a compliance auditing framework built using managed Azure services and open orchestration tools. The key contributions of this work are:

Temporal Evidence Timeline (TET). A unified, time-based representation of the information present in the transcript segments, text overlays and visual labelling. This enables all claims to be linked to a specific point in time.

Cross Modal Contradiction Detection (CMCD). This module links the claims to surrounding disclaimers and calculates a Disclosure Adequacy Score based on proximity, length, readability and logicality.

Jurisdiction-aware CRAG. A retrieval system which scores the retrieved rules based on relevance, authority, jurisdiction and recency. Poor queries will be rewritten and the system will refuse to reason if there is not enough evidence [14].

Three-way output and escalation. Results will be either 'Pass', 'Fail' or 'Needs Human Review'. If there is any uncertainty, the issue will be escalated.

Auditability. A state machine built using LangGraph with tracing provided using LangSmith. This means all results can be replicated and explained [6, 7].

The remainder of this paper is structured as follows: Related Work is outlined in section 2, our proposed framework in section 3 and implementation in section 4. The experimental design is provided in section 5, followed by a worked example and expected results in section 6. Ethical considerations are discussed in section 7 and finally concluded in section 8.

2. RELATED WORK

2.1 Detection of Violations in Advertising Videos

RAVEN [11] incorporates curriculum reinforcement learning and a multimodal language model to utilise Group Relative Policy Optimisation to train to detect categories of violations and identify their timings. RAVEN++ improves upon this method by using active reinforcement learning and hierarchical rewards to detect fine-grained violations. This has been deployed via online A/B testing [10]. BLM Guard incorporates chain of thought reasoning and policy principles to account for discrepancies between modalities in short video advertising [2]. R³ further adds text rewriting for violations in speech transcripts and on-screen text whilst attempting to maintain the advertisers intent [1]. All four of these methods require large proprietary data sets with training the models being the main obstacle to deployment outside of large platforms.

2.2 MMCM Using LLMs

In a study by Levi et al. [12] GPT-4o, Gemini and Llama were tested on 1500 brand safety videos. The models were significantly cheaper to implement than humans but were still less accurate for nuanced cases. This lead to the recommendation of using a hybrid approach. Al Dahoul et al. [18] observed that general language models outperformed previous moderation models for text, image and video. Cross-modal similarity representations have been shown to improve performance over single modality representations for propagating LLM decisions through Google Ads [17]. This suggests that a multi-modality approach should be used. Additionally, having humans involved in the process would be beneficial.

2.3 Retrieval Augmented Generation

RAG uses external sources of information [24]. Self-RAG trains the model to know when to search for information and includes criticism of the output [15]. CRAG adds a light-weight retrieval critic. This will add labelling to the sources it finds with the labels Correct, Incorrect and Ambiguous. If the label is Incorrect it will refine the search [14]. This is very important for compliance as if the wrong clause is cited this could be worse than not having any verdict. RAGAS can be used to calculate reference free metrics such as faithfulness and context relevance [16].

2.4 Evidence Extraction, Orchestration and Observability

Azure AI Video Indexer uses over thirty different models to analyse uploaded media. It produces JSON outputs containing insightful information about the media. This includes transcripts and OCR outputs with timestamps [3]. The OCR uses the Azure Vision Read model, analysing samples of the video. This will identify both printed and handwritten text in various languages [4, 5]. LangGraph creates graph representations of agent processes with shared states, conditional edges and loops [6]. LangSmith tracks tracing, latency and token usage [7]. Temporal grounding benchmarks such as QVHighlights provide standard metrics for comparing predicted and real time windows [22].

2.5 Existing Open Source and Commercial Tools

Overall the concept of using transcripts, OCR, searching and a language model to verify video adverts is not new. There is an open source project that uses Azure Video Indexer, Azure AI Search, GPT-4o, LangGraph and LangSmith to generate structured violation reports for video content [9]. VeriAd does not aim to be novel within the pipeline. It is innovative in what techniques are added to the pipeline. This includes tiered corrective retrieval using jurisdiction profiles, timed claim and disclaimer matching, quantified disclaimer adequacy and a three way verdict with human escalation.

2.6 Research Gap

Looking at representative examples in table 1 three research gaps can be identified. Firstly, there is rarely a check on the quality of the rules found during retrieval-based compliance pipelines. Secondly there are few systems that will validate whether a disclaimer is adequate for the accompanying claim. Thirdly most outputs are binary so there is no way to handle ambiguous cases. VeriAd has been designed to address these three areas.

Table -1: Comparison of Representative Approaches

Work	Modalities	Rule Retrieval	Time Output	Training Need
RAVEN [11]	Video, audio, text	No	Yes	Large RL training
RAVEN++ [10]	Video, audio, text	No	Yes	Large RL training
BLM-Guard [2]	Visual, speech, subtitles	Policy rules in reward	Partial	SFT and RL
R³ [1]	Speech, on-screen text	No	Text level	RL
Levi, et al. [12]	Frames, transcript, text	No	No	None (zero-shot)
Open-source Azure checker [9]	Transcript, OCR	Agentic RAG	Not reported	None
VeriAd (proposed)	Transcript, OCR, labels	Graded CRAG loop	Yes (TET)	None (API-based)

3. PROPOSED SYSTEM

3.1 Overview

The VeriAd system takes as input an advertisement video and produces a report on its compliance which is backed up by evidence. It includes seven stages: video ingestion, multimodal feature extraction, timeline creation, claim/disclaimer extraction, rule acquisition for correction, cross modal conflict detection and a verdict stage (which optionally includes human verification). The general system is shown in Figure 1.

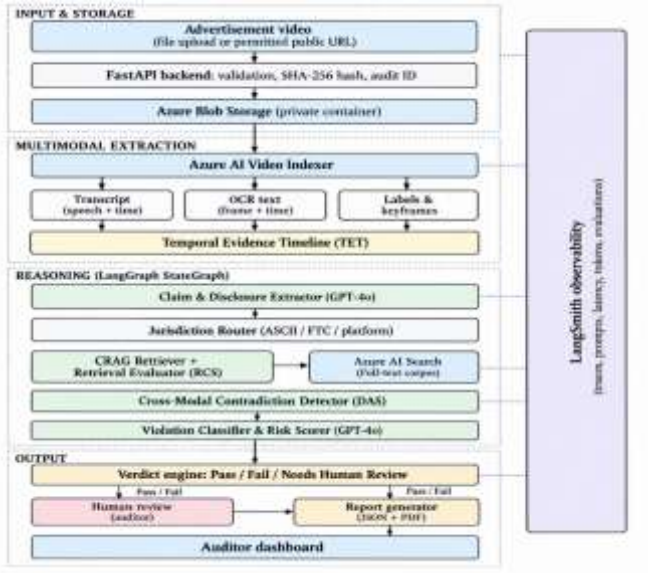


Fig -1: Overall Architecture of VeriAd

3.2 Input and Ingestion

A video file is uploaded or a public URL provided. The primary method of ingestion is via uploading a video file due to the terms of service of the platform preventing reliable retrieval of a video from a URL. Ingestion via URL will only be available for sources which can be downloaded. The video file will then be validated for the type of file, size and length. It will then be hashed using SHA-256 to ensure a clip isn't processed multiple times. This will be stored in a private Azure Blob Storage container. At this stage a unique audit ID will also be generated.

3.3 Multimodal Extraction and Temporal Evidence Timeline

The video will then be sent to Azure AI Video Indexer and return data including lines of transcription, OCR, labels, keyframes and scene changes. All data will have a start time, end time and confidence value [3, 4]. The data is then normalised into one list of evidence items in the format:

$$e = (id, m, text, t_s, t_e, conf, meta) \quad (1)$$

Where, m is the modality (Speech, OCR, Visual Label), t_s is the start time (seconds), t_e is the end time (seconds), conf is the confidence of the extractor and meta contains any further attributes (OCR bounding box, detected language). This will be sorted by t_s to make the Temporal Evidence Timeline. OCR lines will be merged if they are repeated between frames. This means the duration is combined and allows the system to know how long the disclaimer was displayed on screen. Items below a certain confidence threshold will still be included but marked so the reasoning stage does not solely depend on these claims.

3.4 Claims and Disclaimers Extraction

The timeline will be read by GPT-4o [19], with overlapping windows. Structured JSON will then be returned. Claims contain information about the claim, including text, unit IDs, time period, claim type (what type of claim it is) and risk cues (absolute words such as guaranteed, 100%, health results, price promises). Disclaimers contain text, time period, type of disclaimer, font size and language. The model has been prompted to return text exactly as it appears in the evidence so there is no paraphrasing. This ensures claims are honestly cited later. If the JSON is invalid it will be rejected by a schema validator. The node can retry once. All claims will be mapped to the violation taxonomy in Table 2.

Table -2: Violation Taxonomy Used by VeriAd

Code	Category	Typical Example
V1	Unsubstantiated efficacy or health claim	"Cures acne in 3 days"
V2	Absolute or exaggerated claim	"100% guaranteed results"
V3	Missing or inadequate disclaimer	Disclaimer shown for 0.8 s
V4	Misleading price or offer	"Free" with hidden charges
V5	Undisclosed paid endorsement	Influencer ad with no "Ad" tag
V6	False urgency or scarcity	"Only 2 left" on a permanent offer
V7	Unsupported comparative claim	"Better than Brand X"
V8	Environmental claim without basis	"Eco-friendly" with no proof
V9	Cross-modal contradiction	Voice says "risk-free", text says "subject to market risk"

3.5 Jurisdiction Router and Corrective Retrieval

Different rules apply in different markets. A router will select one or more regulatory profiles. These can be: India (ASCI Code, CCPA guidelines) [13, 21], United States (FTC Endorsement Guides and disclosure guidance) [20, 25] and European Union (Unfair Commercial Practices Directive) [26]. We also include platform's advertising policies. The profile selected will depend on what the user has chosen. The language and currency will simply be a hint. If there are multiple profiles in use, then a verdict will be returned for each one. If they do not agree, they will be displayed. Our knowledge base consists of multiple versions of rule clauses. Each rule clause has information about the source, section number, jurisdiction, effective date and authority. This data is stored within Azure AI Search so we can perform hybrid keyword and vector search [8].

A query was created using the claim, type and active profile. Top k rule clauses were retrieved. Each rule clause d was scored using the Retrieval Confidence Score shown in Equation (2).

$$RCS(q,d) = \alpha\, S_rel + \beta\, S_auth + \gamma\, S_jur + \delta\, S_rec \quad (2)$$

Where, S_rel represents the relevance scored by a grader LLM. S_auth is 1 if it is from a statute or official code, 0.7 if it is an official guideline and 0.4 if it is a platform help page. S_jur is 1 if the rule clause matches the active profile. S_rec decreases based on how old the rule clause is. The weighting of these factors add up to 1 and is trained on the validation split. If the highest scoring rule clause is above a threshold θu then:

- **Correct -** we will store the passed rule clauses and remove sentences that do not relate to the claim.

If the highest scoring rule clause is between a lower threshold θl and higher threshold θu then:

- **Ambiguous -** we will re-write the query using the type and synonyms. Merge the results.

If the highest scoring rule clause is below the lower threshold θl then:

- **Incorrect -** discard the results and re-write the query. If we have re-written twice and still cannot find a correct rule clause, we will mark the claim as having 'insufficient legal context' and send it to be human reviewed.

VeriAd does not utilise open web search like original CRAG as we cannot use unverified web pages as legal sources. We limit how many times we re-write the query to prevent unnecessary costs and latency. The full loop can be seen in Figure 2.

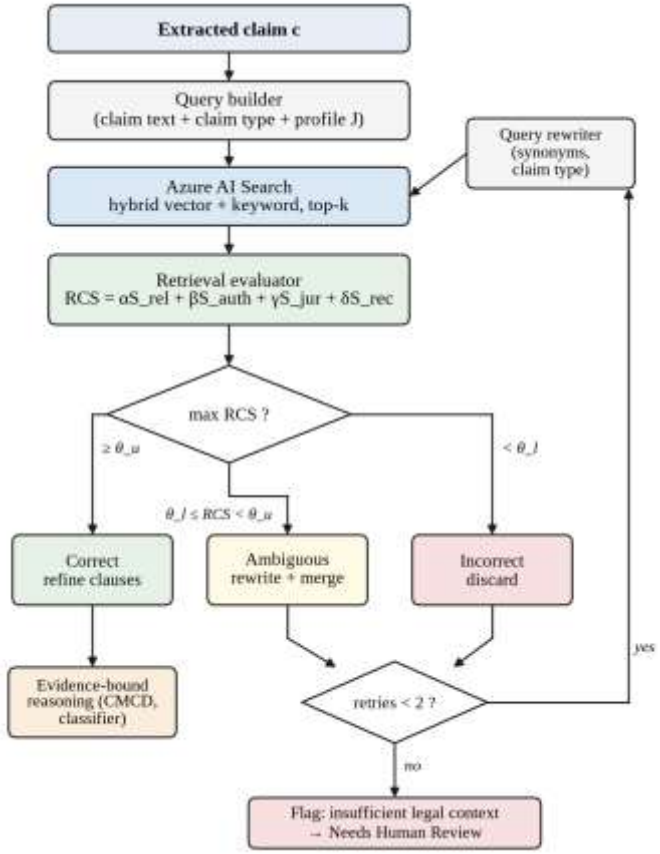


Fig -2: Jurisdiction-Aware CRAG Retrieval Loop

3.6 Cross-Modal Contradiction Detection

This module is the core novelty of VeriAd. For each claim c, every disclaimer u whose time range lies within a window W (15 s by default) of the claim is collected. For each pair (c, u), four signals are computed, the first three of which are given in Equations (3) to (5).

$$P = \exp(-|\Delta t| \div \tau) \quad (3)$$

$$D = \min(1, \text{dur}(u) \div d_min) \quad (4)$$

$$L = \text{conf_OCR}(u) \times \min(1, h(u) \div h_min) \quad (5)$$

Here, Δt is the time gap between the claim and the disclaimer and τ is a decay constant. The term dur(u) is how long the disclaimer stays on screen, and d_min is the minimum readable duration, estimated from the word count at a reading speed of about three words per second. The term h(u) is the text height as a fraction of frame height, and h_min is a minimum legible height. The fourth signal, C, is the probability that u contradicts c. It is produced by a natural language inference model such as DeBERTa [23] and checked by GPT-4o on borderline cases. The Disclosure Adequacy Score is then computed as in Equation (6).

$$DAS(c, u) = w1 \times P + w2 \times D + w3 \times L + w4 \times (1 - C) \quad (6)$$

A claim that needs qualification, but has no disclaimer with a DAS above the threshold θ_d, is flagged as V3. A pair with a high C is flagged as V9, a cross-modal contradiction, even when the disclaimer is clearly visible, because a disclaimer cannot repair a headline claim that it directly contradicts. This follows both the FTC position that a disclosure cannot cure a false claim and must sit close to the claim it qualifies [25], and the CCPA rule that a disclaimer shall not attempt to correct a misleading claim [21]. A disclaimer written in a different language from the claim is also flagged, in line with the CCPA guidelines [21]. The alignment for a sample clip is illustrated in Figure 3.

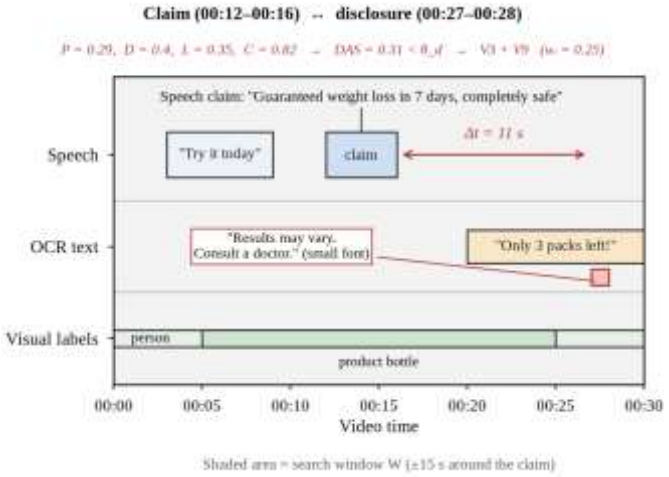


Fig -3: Claim and Disclaimer Alignment on the Temporal Evidence Timeline

3.7 Violation Categorisation, Score and Verdict

GPT-4o is provided with a single claim along with its supporting evidence units, best disclaimer pairs and DAS values, and the refined rules clauses. The model should output a category from Table 2, severity of violation s ∈ {1,2,3}, confidence level p, list of evidence and rule clause IDs used, and recommended remediation. A post-processing step ensures no violation uses any clause or evidence IDs that weren't provided. The overall risk score is calculated using:

$$R = \max_i (s_i \times p_i) / 3 \quad (7)$$

A final verdict of Fail will be given if any violations have s = 3 with p ≥ 0.8 or a risk score greater than or equal to a threshold θ_f. A Needs Human Review will be given if any claims have no legal context, any violation has 0.5 ≤ p < 0.8, or some key evidence has been retrieved via a low confidence OCR source. Otherwise, a verdict of Pass will be returned. These thresholds have been set conservatively to ensure no false failures occur which could cause an advertiser extra time and cost. If there is any uncertainty regarding any violations, these will be passed to a human.

3.8 LangGraph

This workflow can be implemented using a LangGraph StateGraph [6]. Each node has access to and updates a single state object containing the audit ID, video metadata, timeline, claims, disclaimers, scored retrieval results, contradicting pairs, violations, verdict and trace ID. Conditional edges are used to implement the CRAG retry loop and escalation to review. The state is saved between each node using a checkpointer, meaning if the Video Indexer fails or review is paused, the entire task doesn't need to be restarted. The different nodes are shown in Table 3 and the graph shown in Figure 4.

Table -3: LangGraph Nodes and Transitions

Node	Main Output	Next Node
ingest	Blob URI, hash, audit ID	index_video
index_video	Raw Video Indexer JSON	build_timeline
build_timeline	Sorted evidence units (TET)	extract_claims
extract_claims	Claims and disclaimers	route_jurisdiction
route_jurisdiction	Active rule profiles	retrieve_rules
retrieve_rules	Top-k clauses	grade_retrieval
grade_retrieval	RCS and chosen action	rewrite_query, cmcd or human_review
rewrite_query	New query, retry count	retrieve_rules
cmcd	Claim and disclaimer pairs with DAS	classify
classify	Findings with severity and confidence	verdict
verdict	Pass, Fail or Review, risk R	report or human_review
human_review	Reviewer decisions	report
report	JSON and PDF audit report	END

```
Algorithm 1: VeriAd Audit Procedure
Input : video V, regulatory profiles J
Output: audit report Rep
1  uri <- Ingest(V);  X <-
VideoIndexer(uri)
2  T <- BuildTimeline(X)
3  (C, U) <-
ExtractClaimsAndDisclaimers(T)
4  for each claim c in C do
5     k <- 0;  q <- BuildQuery(c, J)
6     repeat
7        D <- Search(q);  a <- Grade(D)
8        if a != Correct then q <-
Rewrite(q); k <- k + 1
9     until a = Correct or k = 2
10    if a != Correct then Flag(c,
NO_LEGAL_CONTEXT)
11    pairs <- CMCD(c, U, W)
12    f_c <- Classify(c, pairs,
Refine(D))
13 end for
14 (verdict, R) <- Decide({f_c})
15 if verdict = Review then HumanReview()
16 return Report(verdict, R, {f_c},
traceID)
```

The control flow of a complete audit is summarised in Algorithm 1.

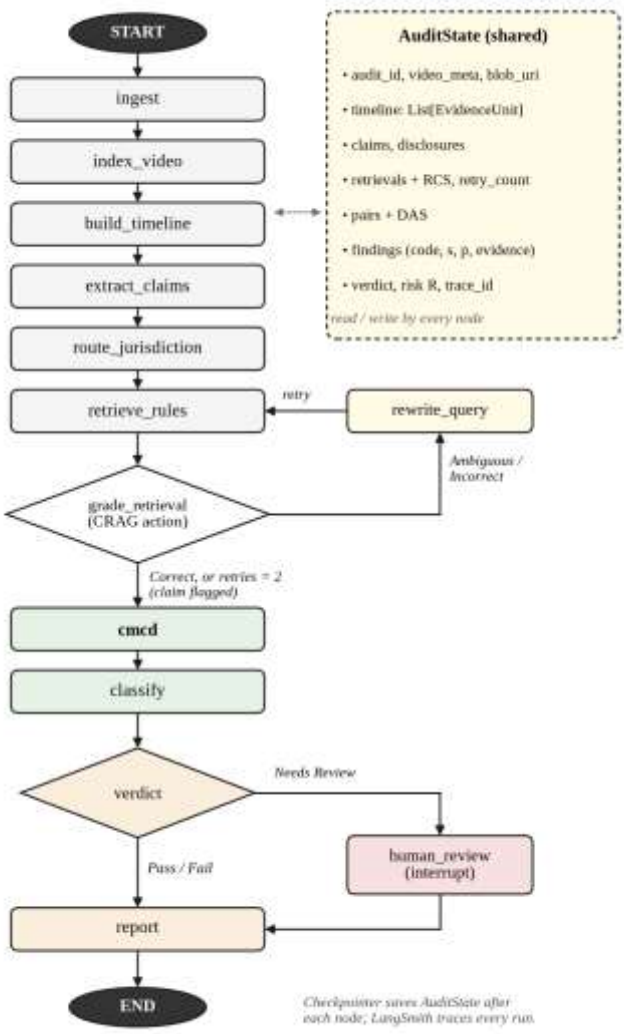


Fig -4: LangGraph State Graph of VeriAd

3.9 Human-in-the-loop

Once a job has been sent for review, the run will stop at the interruption point. The reviewer can then view the dashboard and navigate to the correct time points using the timestamps. Decisions can then be made about whether the findings should be accepted, rejected, edited or if there should be a new rule added. These decisions are stored alongside the associated audit ID. This allows a labelled dataset to be built up over time, allowing thresholds in the equations (2)-(7) to be adjusted and the precision measured in real world scenarios. This aligns with recent studies on moderation, suggesting this hybrid approach should be taken [12].

3.10 Observability

All information regarding the nodes being run within the system such as prompts to the models, responses from the models, results from the retriever, token counts and latencies are stored within LangSmith [7]. All of this information is stored with the audit ID. The LangSmith trace ID, version of the rule corpus and model deployment name are all stored within the final report. This allows anyone wishing to view previous decisions to exactly replicate what the system would have seen. The fields included within the final report are detailed in the Appendix.

4. IMPLEMENTATION

The backend has been implemented using Python with FastAPI. Long running jobs are handled by running them asynchronously with a task queue, meaning that the API will respond quickly with a job ID. The frontend has been created as a React dashboard including the video player, timeline lanes, findings and final report. The key components can be seen in the table below (Table 4).

Table -4: Technology Stack

Layer	Component	Role
Storage	Azure Blob Storage	Private store for videos and reports
Extraction	Azure AI Video Indexer	Transcript, OCR, labels, keyframes [3]
Retrieval	Azure AI Search	Hybrid vector and keyword index [8]
Embeddings	Azure OpenAI embeddings	Vectors for rule clauses
Reasoning	Azure OpenAI GPT-4o	Extraction, grading, classification [19]
Consistency	NLI model (DeBERTa)	Contradiction probability C [23]
Orchestration	LangGraph	Stateful graph, loops, interrupts [6]
Observability	LangSmith	Tracing, evaluation, monitoring [7]
API and UI	FastAPI, React	Job API and auditor dashboard
Security	Azure Key Vault, managed identity	Secrets and access control

4.1 Knowledge Base Creation

All documents were sourced from official rules including ASCI Code [13], CCPA guidelines [21], FTC Endorsement Guides [20], FTC guidance [25], and EU Unfair Commercial Practices Directive [26], as well as platform ad policy documents. Each document was separated along clause lines. This means each chunk contains one citable rule, on average. Metadata was added to the chunks specifying the URL of the source document, clause number, country, authority level and effective date. When the source is updated, a new version is indexed, and the profile is changed. This means that previous audits can still be reproduced using the original rules.

4.2 Prompting and Controlling Outputs

All calls to GPT-4o are made using structured outputs with JSON schema and a low temperature. All prompts specify that the model should only use the given evidence from the provided clauses. If there is not enough evidence then it should respond with "insufficient evidence". This, along with the citation check outlined in Section 3.7 is the main way of preventing hallucinations.

4.3 Security and Privacy

The video files are stored within a private container with temporary access tokens. Secrets are stored in Azure Key Vault and services are authenticated through managed identities. In addition, as these are not required for checking compliance, the face recognition features of Video Indexer have been disabled to avoid any unnecessary privacy issues [5].

5. EXPERIMENT

5.1 Dataset

300-500 short videos will be used for evaluation in 5 different categories; health & wellness, finance & loans, education & coaching, e-commerce promotions and influencer promotions. Videos will be pulled from public ad libraries, upheld complaints from ASCI reports and synthetic ad videos created by the team to capture some rare scenarios such as conflicting disclaimers. Two annotators will annotate each video by labelling all claims and disclaimers with timestamps, assigning the violation code from Table 2, identifying the relevant rule clause and providing an overall verdict. Any disagreements will be resolved by a third annotator, and Cohen's Kappa will be reported to show agreement between annotators. The dataset will be split into validation (20%) and test (80%). Only validation will be used to find ideal threshold and weight values.

5.2 Baselines and Ablation

There are 5 different configurations being tested to understand how each design choice impacts performance, as shown in Table 5. Each step only adds 1 element.

Table -5: Experimental Configurations

ID	Configuration
B1	Transcript only, single-pass RAG, GPT-4o
B2	Transcript and OCR (concatenated), single-pass RAG
B3	B2 with the Temporal Evidence Timeline
B4	B3 with jurisdiction-aware CRAG
B5 (VeriAd)	B4 with CMCD and the three-way verdict

5.3 Evaluation Metrics

Detection task performance is evaluated using precision, recall and F1 score. For a detection to be considered correct, it must match the true finding category, and have a non-zero intersection in terms of time. Temporal IoU [22] is used to evaluate temporal accuracy. This is calculated using Equation (8), where A represents the detected time period, and G represents the ground truth period.

$$tIoU = |A \cap G| / |A \cup G| \quad (8)$$

The retrieval task is evaluated using the Hi@t metric. This measures the proportion of claims that had the correct clause within their top-k results. Citations are evaluated using citation accuracy, which measures the proportion of findings that have a clause cited that does indeed support the decision (as verified by annotators). Faithfulness and contextual relevance are assessed using RAGAS [16]. Verdict accuracy, false failure rate and review rate are provided for the clip level. Verdict accuracy refers to how accurate the verdict was, false failure refers to the proportion of compliant ads that failed, and review rate refers to the proportion of clips that were sent to a human. Finally operational cost is measured using the average latency and number of tokens used per clip, taken from LangSmith traces.

6. RESULTS AND DISCUSSION

6.1 Example Case Study

To demonstrate how each module interacts, consider a 30 second example clip of a weight loss supplement advert with the India profile. This advert has the voiceover saying "Guaranteed weight loss in 7 Days! 100% Safe!" between time

00:12-00:16. There is also a small line of text saying "Results may vary. Consult Doctor." between time 00:27-00:28 at the bottom of the screen. There is also a banner saying "Only 3 Packs Left!" for the duration of 00:20-00:30. An example of what the pipeline would output is shown in Table 6. These values are examples and do not represent actual results.

Table -6: Illustrative Output for the Sample Clip

Time	Evidence	Finding	Basis
00:12 to 00:16	Speech: "Guaranteed weight loss in 7 days"	V1 and V2, severity 3	Claims must be substantiated and not exaggerated [13, 21]
00:27 to 00:28	OCR: "Results may vary"	V3 and V9; $\Delta t = 11$ s, D = 0.4, DAS = 0.31	Disclaimer too late, too brief, and contradicts "guaranteed" [21, 25]
00:14	Speech: "completely safe"	V2, severity 2	Absolute safety claim without evidence [13]
00:20 to 00:30	OCR: "Only 3 packs left!"	Needs review (p = 0.6)	Possible false urgency; stock data unavailable

Overall, the result has failed due to a high confidence critical finding, giving a value close to 1. The low stock flag is not included in the failure as it is provided separately for the reviewer. It is impossible for the system to know the actual stock level. An audio only baseline (B1) would pick up on the 'guaranteed' but would not know when the disclaimer was spoken so would completely miss the point. It would also not be able to see the low stock flag. A concatenation baseline (B2) would see the disclaimer however without knowing when it was spoken it may consider the 'Results may vary' to be enough of a disclaimer, which would cause it to give a less severe result. This is what the CMCD is trying to avoid.

6.2 Quantitative Results

Table 7 shows a template of the results. Once the evaluation set has been created this can be filled in. From the design we expect to see the biggest increase in recall between B1 and B2. This is due to many of the disclaimers and offers being on screen only. B4 should give a higher accuracy of citations and fewer false fails than B3. This is because any weakly related clauses have been filtered out. B5 should help with the detection of V3 and V9. Some clips should be sent for review however the proportion should be small. A lower review rate is not necessarily a good thing if there are more false fails.

Table -7: Results on the Test Set (To Be Completed After Experiments)

ID	P	R	F1	tIo U	Hit @5	C A	FF R	R R
B1	–	–	–	–	–	–	–	–
B2	–	–	–	–	–	–	–	–
B3	–	–	–	–	–	–	–	–
B4	–	–	–	–	–	–	–	–
B5	–	–	–	–	–	–	–	–

Table 7. P - precision, R - recall, CA - citation accuracy, FFR - false-failure rate, RR - review rate.

Figure 5 illustrates the planned auditor interface for the sample clip. This is a mock up and will be substituted with an actual screenshot of the running system once implemented



Fig -5: VeriAd Auditor Dashboard (Conceptual Mockup)

6.3 Limitations

VeriAd is reliant on the quality of the output provided by Video Indexer. Using stylised fonts, quick movement and poor-quality images can impact the accuracy of the OCR and transcription [5]. This framework also works on text extracted from videos meaning it can still be susceptible to visual manipulation such as edited before and after images. The use of GPT-4o to evaluate the grade increases cost and latency, however, this is reduced by limiting the number of rewrites to two and using a hash-based cache. We highlight the risk but do not provide legal advice. The rule set must be maintained by humans.

7. ETHICAL AND PRACTICAL CONSIDERATIONS

Pulling data from social media can violate terms of service, therefore we default to uploading files and only using URLs from allowed sources. We avoid using biometric features, all videos will be private and are able to be deleted. As a Fail can have negative impacts on a business, we show the evidence provided for each finding and what rule was triggered. Any uncertainty can be decided by the reviewer. We keep regulations separated so no rule from another country will be silently applied to an advert.

8. CONCLUSION

In this paper we introduced VeriAd, a multimodal framework for auditing short-form advertisement videos for compliance. Rather than simply sending unstructured text to a language model, VeriAd creates a timeline of evidence based on the timestamped information. Rules are then graded using a jurisdiction aware CRAG loop. Each disclaimer is checked to ensure they are sufficiently similar, lengthy, legible and consistent to qualify the claim. LangGraph ensures we can track the multi-stage process. LangSmith means we can trace all decisions made and we provide a three-way verdict to send any uncertainty to a human reviewer. The whole framework uses only managed services and no model training meaning it is feasible for small teams and agencies.

Future work will include completing the annotated data set and providing the results of our ablation study shown in table 7. New languages will be added for regional languages within India. We will also expand our visual analysis to include before and after images and tampered media. An automated pipeline will be created to notify maintainers if the source regulation changes. Using feedback from a reviewer to automatically adjust the thresholds is also a promising area of work.

9. APPENDIX

A summary of the audit report structure produced by VeriAd for each video is provided in table 8.

Table -8: Fields of the VeriAd Audit Report

Field	Content
audit_id	Unique ID created at ingestion, with the SHA-256 hash of the video
verdict	Pass, Fail or Needs Human Review, with overall risk R
profiles	Active regulatory profiles and the verdict under each one
findings	Category, severity, confidence, time range and suggested fix for each finding
evidence	Exact transcript and OCR text, with timestamps and frame references
citations	Rule clause text, source URL, clause number and corpus version
disclaimer_ scores	P, D, L, C and DAS for every claim and disclaimer pair
review_log	Reviewer decisions, edits and comments, if any
trace	LangSmith trace ID and model deployment name

ACKNOWLEDGEMENT

We would like to thank [Ashish P Mohod] of the Department of Computer Science and Engineering -II at [Priyadarshini Bhagwati College Of Engineering , Nagpur] for guiding us during this project. We also thank the department for providing us all the necessary resources.

REFERENCES

1. Y. Chen, Z. Hu, M. Xue, T. Cao, L. Liu, P. Shu, H. Yu, J. Jiang, "R³: Advertisement Compliance Rectification via Group-Relative Experience Extractor and Curriculum Reinforcement", Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (ACL), Industry Track, 2026. https://arxiv.org/abs/2607.07318
2. Y. Yang, Z. Liu, Y. Yuan, Y. Song, X. Ma, Y. Song, X. Zeng, L. Sun, Y. Wang, H. Zhou, S. Cui, Z. Gong, J. Zhang, "BLM-Guard: Explainable Multimodal Ad Moderation with Chain-of-Thought and Policy-Aligned Rewards", arXiv preprint arXiv:2602.18193, 2026. https://arxiv.org/abs/2602.18193
3. Microsoft, "Azure AI Video Indexer Insights Overview", Microsoft Learn Documentation, 2026. https://learn.microsoft.com/en-us/azure/azure-video-indexer/insights-overview
4. Microsoft, "Get Optical Character Recognition (OCR) Insights", Microsoft Learn Documentation, 2026. https://learn.microsoft.com/en-us/azure/azure-video-indexer/ocr-insight
5. Microsoft, "Transparency Note for Azure AI Video Indexer", Microsoft Learn Documentation, 2026. https://learn.microsoft.com/en-us/legal/azure-video-indexer/transparency-note
6. LangChain, "LangGraph Documentation", LangChain Inc., 2026. https://langchain-ai.github.io/langgraph/
7. LangChain, "LangSmith Documentation", LangChain Inc., 2026. https://docs.smith.langchain.com/
8. Microsoft, "Vector Search in Azure AI Search", Microsoft Learn Documentation, 2026. https://learn.microsoft.com/en-us/azure/search/vector-search-overview
9. GitHub user raghavbankar, "Video Compliance System: Automated Video Compliance QA Using Agentic RAG and LLMs", GitHub Repository, accessed October 2026. https://github.com/raghavbankar/Video_Compliance_Analyzer
10. D. Ji, Y. Yang, L. Liu, P. Shu, H. Wu, S. Tang, X. Chen, S. Ma, T. Chen, L. Zhu, "RAVEN++: Pinpointing Fine-Grained Violations in Advertisement Videos with Active Reinforcement Reasoning", Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing (EMNLP), Industry Track, 2025. https://arxiv.org/abs/2511.19168
11. D. Ji, Y. Yang, H. Wu, S. Ma, T. Chen, L. Zhu, "RAVEN: Robust Advertisement Video Violation Temporal Grounding via Reinforcement Reasoning", Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (ACL), Industry Track, 2025. https://arxiv.org/abs/2510.16455
12. A. Levi, O. Levi, S. Mishra, J. Morra, "AI vs. Human Moderators: A Comparative Evaluation of Multimodal LLMs in Content Moderation for Brand Safety", Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops (ICCVW), 2025, DOI 10.1109/ICCVW69036.2025.00627. https://arxiv.org/abs/2508.05527
13. Advertising Standards Council of India, "The Code for Self-Regulation of Advertising Content in India", ASCI, July 2025. https://www.ascionline.in/the-asci-code/
14. S. Yan, J. Gu, Y. Zhu, Z. Ling, "Corrective Retrieval Augmented Generation", arXiv preprint arXiv:2401.15884, 2024. https://arxiv.org/abs/2401.15884
15. A. Asai, Z. Wu, Y. Wang, A. Sil, H. Hajishirzi, "Self-RAG: Learning to Retrieve, Generate, and Critique through Self-Reflection", Proceedings of the International Conference on Learning Representations (ICLR), 2024. https://arxiv.org/abs/2310.11511
16. S. Es, J. James, L. Espinosa-Anke, S. Schockaert, "RAGAs: Automated Evaluation of Retrieval Augmented Generation", Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (EACL): System Demonstrations, March 2024, 150–158. https://aclanthology.org/2024.eacl-demo.16/
17. W. Qiao, T. Dogra, O. Stretcu, Y. Lyu, T. Fang, D. Kwon, C. Lu, E. Luo, Y. Wang, C. Chia, A. Fuxman, F. Wang, R. Krishna, M. Tek, "Scaling Up LLM Reviews for Google Ads Content Moderation", Proceedings of the 17th ACM International Conference on Web Search and Data Mining (WSDM), March 2024, DOI 10.1145/3616855.3635736. https://arxiv.org/abs/2402.14590

18. N. AlDahoul, M. Tan, H. Kasireddy, Y. Zaki, "Advancing Content Moderation: Evaluating Large Language Models for Detecting Sensitive Content across Text, Images, and Videos", arXiv preprint arXiv:2411.17123, 2024. https://arxiv.org/abs/2411.17123
19. OpenAI, "GPT-4o System Card", arXiv preprint arXiv:2410.21276, 2024. https://arxiv.org/abs/2410.21276
20. Federal Trade Commission, "Guides Concerning the Use of Endorsements and Testimonials in Advertising", Code of Federal Regulations, Title 16, Part 255, July 2023. https://www.ecfr.gov/current/title-16/chapter-I/subchapter-B/part-255
21. Central Consumer Protection Authority, "Guidelines for Prevention of Misleading Advertisements and Endorsements for Misleading Advertisements, 2022", Department of Consumer Affairs, Government of India, June 2022. https://consumeraffairs.gov.in/public/upload/files/CCPA_Notification_1732707665.pdf
22. J. Lei, T. Berg, M. Bansal, "QVHighlights: Detecting Moments and Highlights in Videos via Natural Language Queries", Advances in Neural Information Processing Systems (NeurIPS), 2021, 34. https://arxiv.org/abs/2107.09609
23. P. He, X. Liu, J. Gao, W. Chen, "DeBERTa: Decoding-Enhanced BERT with Disentangled Attention", Proceedings of the International Conference on Learning Representations (ICLR), 2021. https://arxiv.org/abs/2006.03654
24. P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, H. Küttler, M. Lewis, W. Yih, T. Rocktäschel, S. Riedel, D. Kiela, "Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks", Advances in Neural Information Processing Systems (NeurIPS), 2020, 33, 9459–9474. https://arxiv.org/abs/2005.11401
25. Federal Trade Commission, ".com Disclosures: How to Make Effective Disclosures in Digital Advertising", FTC Staff Guidance, March 2013. https://www.ftc.gov/business-guidance/resources/com-disclosures-how-make-effective-disclosures-digital-advertising
26. European Parliament and Council of the European Union, "Directive 2005/29/EC Concerning Unfair Business-to-Consumer Commercial Practices in the Internal Market (Unfair Commercial Practices Directive)", Official Journal of the European Union, May 2005, L 149, 22–39. https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX:32005L0029