Multi-Agent Explainable Video Interview Evaluation

Yousef Khojah, Mohammed Alnasser

Mentor: **Ahmed Radwan**

yousef.khojah@kaust.edu.sa, Mohammed.alnasser@kaust.edu.sa

ahmedyradwan02@gmail.com

Abstract

Recorded, asynchronous interviews are now common in hiring and interview-practice settings, and automated tools increasingly score them, often reducing an answer to a single, unexplained number. This project builds a trustworthy, explainable, multi-agent system that evaluates a recorded interview answer and returns criterion-level scores, timestamped evidence drawn from the transcript, an explicit verification of every claim, a calibrated confidence value, and concrete feedback for improvement. The guiding question is whether a structured multi-agent multimodal system can evaluate interview answers more transparently and usefully than a single end-to-end multimodal system given identical inputs. The system is implemented end to end: a preprocessing pipeline turns a raw interview video into extracted audio, a word-aligned timestamped transcript, and sampled frames; a model-agnostic provider abstraction supports both self-hosted and API-hosted vision-language models; and two comparable pipelines, a single-call multimodal baseline and a multi-agent workflow, emit an identical structured evaluation report so their outputs can be compared directly. Both pipelines were evaluated on 75 RecruitView samples. The multi-agent pipeline attaches timestamped evidence to 78 percent of rubric criteria and flags unsupported claims in 55 percent of samples, outputs the baseline does not produce at all. The baseline scored slightly higher on reference-similarity and LLM-judge measures at roughly one third of the runtime. Two findings qualify that gap: once the multi-agent system's own reported scores and confidence values are excluded, both systems make claims about the candidate at statistically indistinguishable rates, and every reference summary in the dataset performs the personality and emotional-state inference that this system is designed to exclude.

Date: August 23, 2026

Contents

1 Introduction **2**

1.1 Motivation . . . 2

1.2 Problem Statement . . . 2

1.3 Project Scope . . . 3

1.4 System Summary . . . 3

2 Related Work **3**

3 Methodology **4**

3.1 System Overview . . . 4

3.2 Preprocessing Pipeline . . . 5

3.3 Model Provider Abstraction . . . 6

3.4 Evaluation Rubric . . . 6

3.5 Agent Architecture . . . 7

3.6 Deterministic Assembly . . . 7

3.7 Explainability and Trustworthiness Safeguards . . . 8

3.8 Evaluation Harness . . . 8

4 Results **9**

4.1 Experimental Setup . . . 9

4.2 Example Output . . . 9

4.3 Structural Output . . . 9

4.4 Summary Quality Against Reference Summaries . . . 10

4.5 Judge-Flagged Claims . . . 10

4.6 Timestamp Verification . . . 10

4.7 Reference Summary Characteristics . . . 11

4.8 Cost . . . 11

4.9 Evidence Span Localisation . . . 11

5 Discussion **12**

6 Limitations **12**

7 Future Work **13**

8 Conclusion **14**

References **14**

1 Introduction

1.1 Motivation

Interviews conducted asynchronously, where a candidate records a video response to a fixed question with no interviewer present, have become a standard stage in many hiring and interview-practice workflows. Because reviewing large numbers of long recordings by hand is slow, automated systems are increasingly used to pre-screen or score these responses. In practice, however, this automation tends to reduce a candidate's answer to a single, largely unexplained score. The problems that motivate this project's design are:

- Long interview videos are time-consuming to review, for both automated systems and human graders.
- The shift from face-to-face interviews to recorded video removes the natural feedback loop a candidate would otherwise get from a live interviewer.
- AI-generated decisions are often ambiguous: a score is produced with no indication of what evidence supports it or why it was assigned.
- Participants receive no specific, actionable suggestions for how their answer could have been improved.

These problems matter because an opaque score is of limited use to the person being evaluated, and because a system that cannot show its evidence is difficult to trust or to audit for errors and bias. A score with no evidence is unfalsifiable: a correct assessment and a fluent guess are indistinguishable to anyone who cannot check the reasoning. Recent multimodal models sharpen this problem rather than resolving it, since they produce confident, well-written assessments whose reasoning cannot be traced back to anything in the recording. This motivates a system whose primary output is not a score by itself, but a score together with the evidence and reasoning that produced it.

1.2 Problem Statement

Given a recorded interview answer, audio and video, and the interview question it responds to, the system produces:

- An evaluation of the answer's quality.
- Criterion-level scores rather than a single aggregate number.
- Citations to transcript evidence, localised with timestamps.
- An explanation of how the cited evidence supports each score.
- A verification of whether each claim is actually supported by the evidence cited for it.
- A calibrated estimate of uncertainty, together with warnings for any criterion lacking evidence.
- Specific, actionable feedback for improving the answer.
- A readable prose summary of the assessment.

This yields the main research question: *can a structured multi-agent multimodal system improve the grounding, verifiability, and usefulness of interview evaluation compared with a single multimodal system that performs the same task end to end?* The system is intended for research and decision support, not for making autonomous hiring decisions.

1.3 Project Scope

The system works with English-language interview responses, processed offline from already-recorded video, using audio, video, the interview question, and a derived transcript as input. RecruitView [1] is the dataset used throughout. Explainability, timestamped evidence, claim verification, uncertainty reporting, and improvement feedback are first-class outputs, and the design directly compares a single multimodal pipeline against a multi-agent pipeline on the same inputs.

The rubric is deliberately restricted to observable, question-related criteria. The system does not infer personality, emotion, confidence, honesty, attractiveness, accent, or any demographic or protected characteristic. This constraint is written into every agent prompt rather than being applied as a post-hoc filter.

1.4 System Summary

The system is organised as a preprocessing stage feeding two comparable evaluation pipelines that share a single output schema.

The **single-model baseline** sends the question, timestamped transcript, and sampled frames to a vision-language model in one call and receives a prose summary directly back.

The **multi-agent pipeline** routes the same inputs through four agents, one deterministic assembly step, and a terminal summarisation agent, coordinated by an orchestrator that reads and updates a shared pipeline state.

Both pipelines emit an identical `EvaluationReport` structure, which is what makes a direct comparison between them possible, since any difference in scores, evidence, or feedback can be attributed to the architecture rather than to differences in output format or evaluation criteria.

2 Related Work

The primary data source is RecruitView [1], a dataset of 2,011 English-language interview-answer videos from more than 300 participants, each paired with audio, video, transcript, the interview question, and evaluation labels. The dataset also ships model-generated reference summaries, which are used in this project as the comparison target for both pipelines. Its scale and multimodal, question-aligned annotations make it well suited both for validating a multimodal evaluator and for benchmarking against reference assessments.

For timestamped, localised evidence the system uses WhisperX [2], chosen because it forces word-level and segment-level alignment onto the transcript, a property the evidence citations depend on directly.

Two recent systems are closest to this work in method. Lo et al. [7] present a context-aware and explainable multi-agent framework for resume screening, decomposing the task across four agents responsible for extraction, evaluation, summarisation, and score formatting, with retrieval-augmented generation supplying external context such as industry expertise and certifications. The decomposition closely parallels the one adopted here, but the system operates on text resumes rather than recorded video, and its explainability derives from transparency about what external knowledge informed a judgement rather than from grounding claims in a locatable part of the source.

Sun et al. [8] present CoMAI, a collaborative multi-agent framework for interview evaluation, comprising four agents specialised in question generation, security, scoring, and summarisation, coordinated through a centralised finite-state machine, and applied to a university admissions setting. CoMAI reports rubric-based structured scoring, reasoning traces, and timestamped agent

logs, and it is the closest published architecture to the one built here. Two differences are relevant. First, CoMAI operates on text-based conversational interviews; multimodal input is listed as future work. Second, its timestamps and audit trail record what the agents did, in the form of trace identifiers and immutable session records, rather than locating evidence within a candidate's answer. Its scoring agent performs answer verification in the sense of checking the factual correctness of a candidate's response, not in the sense of checking whether the system's own justification is supported.

Taken together, these systems provide *process transparency*: reasoning traces, agent logs, and audit trails that record what the system did. What none of them provides is *claim-level accountability*, meaning an explanation anchored to a specific locatable moment in the source together with an automatic check that the evidence actually supports the claim. This distinction is the gap the present work addresses.

For the video branch, the system is designed around Qwen3-VL-8B [3], Alibaba's open-weight vision-language series, served locally through vLLM [4], an inference engine built for high-throughput, memory-efficient serving with an OpenAI-compatible API. Hosted alternatives including OpenAI's and Google's Gemini multimodal APIs were also weighed; the deciding factors were cost at the scale needed to run experiments, reproducibility across repeated runs, and data governance given RecruitView's academic, non-commercial terms, set against the quality and mature structured-output support that hosted models currently offer. A model-agnostic provider abstraction sits between the pipeline and the model so that either family can be used without changing the rest of the system.

The multi-agent workflow is coordinated by an orchestrator that reads and updates a shared pipeline state and decides which agent runs next. LangGraph [5] and CrewAI [6] were both considered for this layer, with LangGraph preferred for its explicit state handling and fine-grained control over agent flow, retries, and stopping conditions. The orchestration layer implemented here follows the same pattern directly rather than adopting either framework as a dependency.

Interpretation of attention weights follows the caution established by Jain and Wallace [9] and the subsequent response by Wiegreffe and Pinter [10]: attention indicates where signal flowed through a model, and does not by itself establish that a region determined the output.

3 Methodology

3.1 System Overview

The pipeline takes an interview question together with the corresponding audio and video recording and produces a structured evaluation report. Preprocessing turns the raw recording into a timestamped transcript, extracted audio, and sampled frames. Two comparable pipelines then consume the same preprocessed inputs.

The single-model baseline sends the question, transcript, and sampled frames to the vision-language model in one call and receives a prose summary directly back. It emits no structured evidence.

The multi-agent pipeline routes the same inputs through the following stages:

1. **Evidence Agent**, which retrieves timestamped transcript spans and frame indices relevant to each criterion and assigns no scores.
2. **Quality Agent**, which scores each rubric criterion using the retrieved evidence.
3. **Assembly**, a deterministic code step that joins evidence spans to scores, caps uncited scores, and computes confidence. It makes no model call.

4. **Verifier Agent**, which checks whether each score is supported by the evidence cited for it and lists any unsupported claims.
5. **Feedback Agent**, which produces specific, actionable, STAR-oriented improvement suggestions.
6. **Summary Agent**, which renders the completed report as prose. It runs once, after the orchestration loop has finished, and is never routed to.

An **Orchestrator Agent** observes the shared pipeline state and decides which stage runs next until it emits a finish action. A fixed-order mode executes the same stages in dependency order without routing calls and serves as the control condition. Both modes produce an identical report. Figure 1 shows the resulting control flow.

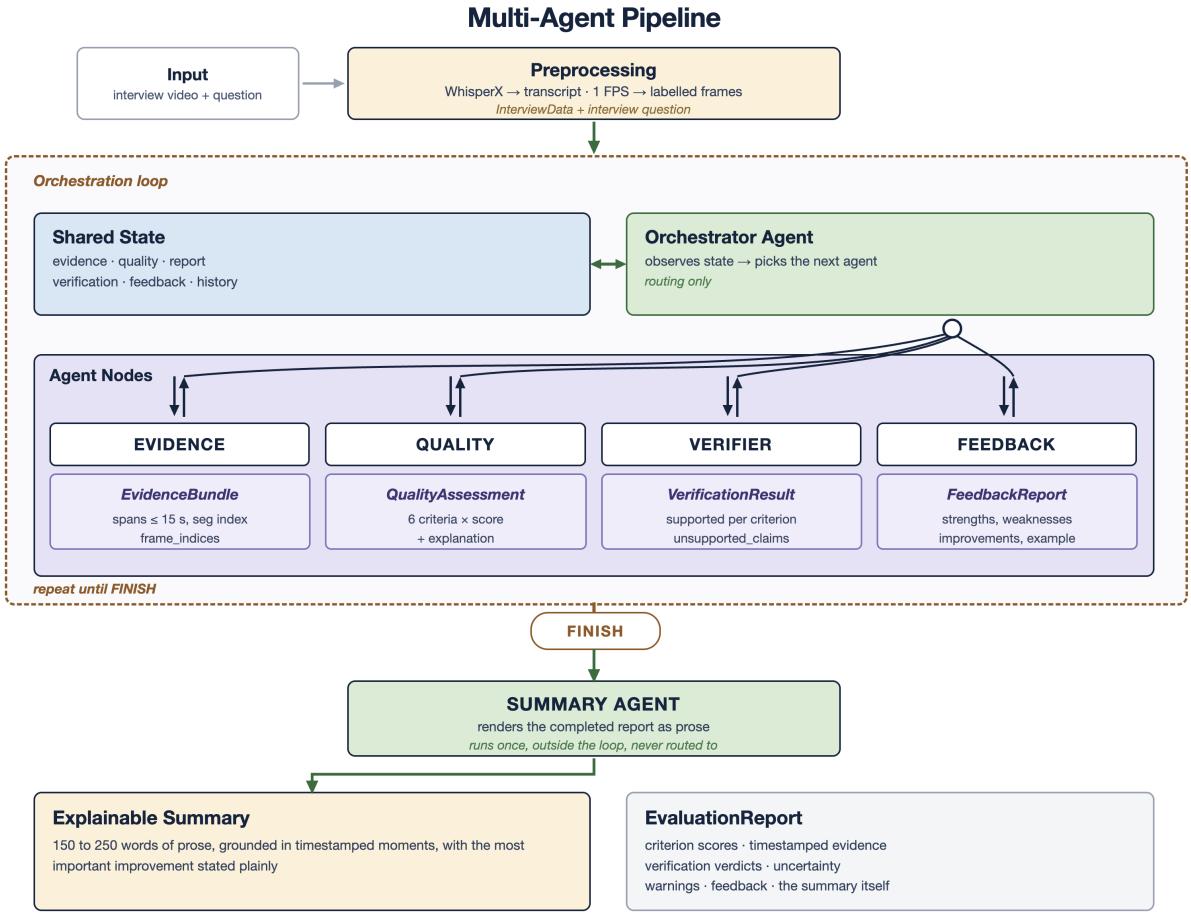


Figure 1: The multi-agent pipeline. Preprocessing produces the transcript and labelled frames; the Orchestrator Agent then reads the shared state and routes to one agent node at a time, each node writing its typed result back to that state, until it emits a finish action. The Summary Agent runs once outside the loop and is never routed to. The deterministic assembly step described in Section 3.6 operates on the shared state rather than as a routed node.

3.2 Preprocessing Pipeline

The preprocessing step turns a raw interview recording into the three artifacts every later component reads from. Given a video file, the pipeline produces:

- Extracted audio, split out from the video's audio track using `ffmpeg` at 16 kHz mono for downstream transcription.
- A transcript, generated by WhisperX [2] with word-level and segment-level timestamp alignment.
- Sampled frames, extracted at a fixed rate of one frame per second, each tagged with its own timestamp.

One frame per second was chosen because it is the simplest uniform policy, and because it matches the default sampling rate used by at least one of the hosted multimodal APIs considered for the video-understanding component. Frames are labelled with their original index and timestamp rather than their position within any subsample, so that a cited frame resolves to a real moment in the video that a reviewer can locate.

Timestamp alignment is a hard requirement here, since it is what allows the system's explanations to point to a specific moment in the recording rather than to the recording as a whole. The transcript is rendered for the model in the form `[seg 12 | 41.2s to 47.8s]`, and frames as `[Frame 34 @ 34.0s]`, so that any timestamp the model cites is one it was shown rather than one it invented.

3.3 Model Provider Abstraction

The preprocessing output connects to the model through a thin, model-agnostic provider interface that takes a complete prompt and the preprocessed interview and returns a parsed structured response. Four providers are implemented: a self-hosted Qwen3-VL-8B provider served through vLLM, a Google Gemini provider, an OpenAI provider, and a mock provider that returns schema-correct canned output for every agent so that the full pipeline can be exercised on a machine with no GPU and no API access.

Each agent declares an `include_frames` flag, so agents that reason over text alone are not charged the cost of image tokens they do not need. In the current configuration the Evidence and Quality agents receive frames; the Verifier, Feedback, and Summary agents do not.

The Gemini provider calls the REST endpoint directly rather than through the vendor SDK. This was necessary because the SDK transmits API keys in the current format as OAuth bearer tokens, which the endpoint rejects, and it has the secondary benefit of removing any dependency on SDK versioning. All providers share a single tolerant JSON parser that strips markdown fences, discards preamble text, and repairs trailing commas before failing, since each malformed response would otherwise cost a real API call to retry.

3.4 Evaluation Rubric

The rubric focuses on criteria that are observable and directly tied to the question asked, rather than on inferred internal states of the candidate. It is encoded as a single six-criterion module shared by both pipelines, so the baseline and the multi-agent system are always scored against the same definitions:

- **Relevance**: whether the response directly answers the interview question.
- **Structure**: logical organisation and flow of ideas.
- **Specificity**: whether concrete examples and details are provided.
- **Completeness**: whether all important aspects of the question have been addressed.
- **Communication clarity**: how clearly and understandably the response is communicated.

- **Reflection**: whether lessons learned or outcomes are discussed where appropriate.

The rubric deliberately excludes emotion, eye contact, attractiveness, accent, personality, honesty, and any protected characteristic, since these are not reliably or fairly assessable from a recorded interview answer. The rubric module is injected into every prompt that needs it rather than being restated in each one, so that all agents score and check against identical criterion names.

3.5 Agent Architecture

Each agent has a distinct epistemic role, and the separation is deliberate rather than decorative:

- **Evidence Agent**: retrieval. Answers what evidence exists, and assigns no scores. Each span must derive from one or at most two adjacent transcript segments and must not exceed fifteen seconds, so that evidence points a reviewer at a specific moment rather than at the answer as a whole. Criteria with no supporting evidence are reported explicitly rather than being assigned weak evidence.
- **Quality Agent**: judgement. Scores each rubric criterion against the retrieved evidence and reports its own certainty, which is distinct from any judgement about the candidate's confidence.
- **Verifier Agent**: audit. Checks whether the cited span exists, whether it contains what is claimed, and whether the explanation asserts anything the evidence does not show. It never changes a score.
- **Feedback Agent**: advice. Produces actionable coaching grounded in what the candidate actually said, and avoids building strong recommendations on criteria the Verifier flagged.
- **Summary Agent**: rendering. Converts the completed report into prose, using only what the report contains.

The Orchestrator Agent never evaluates anything itself. It observes the shared state and routes.

3.6 Deterministic Assembly

An earlier design placed an additional LLM agent between the Quality Agent and the Verifier, responsible for synthesising scores with evidence. That agent was removed once its work proved to be mechanical in every respect. Joining evidence spans to criteria is a lookup, because the Evidence Agent already tags each span with the criterion it supports. Capping an uncited score is a rule. Explanations come directly from the Quality Agent. The step also duplicated the Verifier's function, since both were checking whether scores were evidence-supported, with no rule for which took precedence.

Replacing it with deterministic code removed one model call per sample, removed the duplication, and made scoring and confidence exactly reproducible. Confidence is computed rather than asserted:

$$\texttt{confidence} = \texttt{evidence coverage} \times \texttt{scorer certainty}$$

where evidence coverage is the fraction of rubric criteria carrying at least one retrieved span. A report in which nothing is cited cannot come out confident regardless of how certain the model claims to be. Any criterion with no evidence has its score capped at 2 out of 5, and the cap is logged with its before and after values so that the adjustment is auditable.

3.7 Explainability and Trustworthiness Safeguards

Every criterion's output follows the same chain from observation to actionable feedback:

```
Observed Evidence → Rubric Mapping → Score → Explanation → Verification →
                                Improvement Feedback
```

The safeguards enforced around this chain are:

- Timestamped transcript segments and labelled frames are fed directly to the model as grounding context, so citations are grounded rather than invented.
- A clear separation between what was observed and how the model interpreted it.
- Automatic detection of unsupported claims, performed by the Verifier Agent against the transcript.
- Explicit warnings for any criterion with no retrieved evidence, and automatic score capping for such criteria.
- A confidence value derived from evidence coverage rather than asserted by the model.
- Fairness guardrails written into every prompt: no inference of personality, emotion, confidence, or appearance, and no demographic inference.
- Retry handling and schema validation at the orchestration layer, so that a single malformed response does not lose a run.
- No autonomous hiring decision.

The system returns concise, evidence-based reasoning summaries rather than exposing a model's private chain of thought.

3.8 Evaluation Harness

The benchmark runs in three resumable phases, with per-sample checkpointing, so that a failure costs one sample rather than the whole run:

1. **Predict**: both arms produce summaries for each sample, with per-arm wall-clock timing and model-call counts recorded.
2. **Judge**: an LLM judge scores each summary against the dataset reference summary on helpfulness, completeness, faithfulness, and hallucination, and lists each hallucinated claim verbatim so the count can be audited. The judge is blind to which arm produced a given summary.
3. **Tables**: metrics are computed offline and results tables written.

The offline metrics implemented are ROUGE-L [11], CIDEr [12], cosine similarity, and temporal intersection-over-union with mean IoU and Recall at fixed thresholds. Dataset access is mediated by a manifest file, so that dataset licensing and column naming are handled in one place and the harness can be developed without the dataset present.

4 Results

4.1 Experimental Setup

All results cover 75 RecruitView samples evaluated with the `gemini-3.1-flash-lite` model. Both pipelines received identical inputs: the interview question, the sampled frames, and the word-aligned timestamped transcript. The multi-agent pipeline was run in fixed-order mode. The reference summaries used as the comparison target are the model-generated summaries shipped with the RecruitView dataset.

The Qwen3-VL-8B configuration is implemented and the provider is complete, but the evaluation reported here was not run on it. The results below therefore reflect a small hosted model rather than the intended open-weight deployment.

4.2 Example Output

For the question "What are your greatest strengths and weaknesses?", the system returned an overall score of 3.0 out of 5 with a confidence of 0.60. Relevance scored 5 out of 5 citing 0.0s to 4.5s, structure 4 out of 5 citing the same span, and completeness 3 out of 5 citing 13.7s to 15.6s; all three were marked supported. Communication clarity scored 5 out of 5 citing 0.0s to 4.5s and was flagged, with the Verifier noting that the claim was a subjective assessment not supported by the transcript text. Specificity scored 1 out of 5 and reflection 0 out of 5, both with no evidence retrieved and both flagged, and both were capped by the assembly step.

The prose summary propagated the uncertainty into user-facing text, stating that the assessment of specific examples, communication style, and lessons learned was based on insufficient evidence, and reporting a confidence of 0.6. The single-model baseline on the same sample concluded with the sentence "I am highly confident in this assessment."

4.3 Structural Output

Property	Baseline	Multi-agent
Evidence spans per sample	0	4.72
Rubric criteria carrying evidence	0 of 6	4.69 of 6 (78 percent)
Samples with at least one cited span	0 of 75	73 of 75
Automatic verification pass	none	6 criteria per sample
Samples where the verifier flagged unsupported claims	n/a	41 of 75 (55 percent)
Uncited scores automatically capped	n/a	9 criteria
Confidence signal	none	0.72 ± 0.15 (0.00 to 0.83)

Table 1: Structural output produced by each system per sample.

The baseline values are zero by construction rather than by performance: the single-call pipeline has no mechanism for producing evidence spans, verification verdicts, or a confidence value.

4.4 Summary Quality Against Reference Summaries

System	ROUGE-L	Cosine	CIDEr	Helpful.	Complete.	Faithful.
Baseline	0.088	0.612	0.236	4.77	3.80	4.56
Multi-agent	0.079	0.567	0.134	4.60	3.43	4.11

Table 2: Mean similarity to reference summaries and mean LLM-judge ratings, 75 samples.

CIDEr-D as normally configured returns exactly 0.000 for both arms and is not reported as a result. Its length penalty collapses to zero when reference summaries average 878 tokens against 178 for predictions, which reflects the length ratio and not the content. The CIDEr values above are computed with the length penalty disabled.

Metric	Difference	95% CI	Multi-agent better	Baseline better	p
Helpfulness	-0.17	$[-0.32, -0.04]$	5	15	0.041
Completeness	-0.37	$[-0.53, -0.21]$	6	31	< 0.001
Faithfulness	-0.45	$[-0.68, -0.25]$	8	28	0.001

Table 3: Paired per-sample differences (multi-agent minus baseline), bootstrap 95 percent confidence intervals and sign tests.

4.5 Judge-Flagged Claims

Measure	Baseline	Multi-agent	p
Raw judge-flagged claims per sample	0.80	1.36	< 0.001
of which: own score or confidence value	40 percent	60 percent	
of which: evidence-sufficiency commentary	0 percent	7 percent	
Substantive content claims per sample	0.48	0.45	0.585

Table 4: Judge-flagged claims, decomposed by category.

All 162 flagged claims were categorised. The majority of the multi-agent system's flagged claims are statements such as "achieving an overall score of 3.67" and "my confidence in this assessment is 0.83". The reference summaries contain no scores and no confidence values, so the judge marks these as unsupported. Excluding both that category and evidence-sufficiency commentary, the two systems make claims about the candidate at statistically indistinguishable rates.

4.6 Timestamp Verification

Timestamps cited in each summary were extracted and checked directly against the WhisperX transcript, with no model involved in the check.

System	Timestamps per sample	Within duration	Segment-aligned (± 1.5 s)
Baseline	4.2	99.7 percent	97.1 percent
Multi-agent	4.7	100.0 percent	97.6 percent

Table 5: Timestamp verification against the transcript, computed without any model.

The paired difference in alignment rate is 1.3 percentage points, sign test $p = 0.75$, which is not significant. Given a timestamped transcript in its prompt, the single-model baseline cites real timestamps about as reliably as the multi-agent pipeline does. The difference between the arms lies in whether those citations are attached to specific rubric criteria in a machine-checkable structure, not in whether the timestamps themselves are real.

4.7 Reference Summary Characteristics

Property	Reference summaries	System output
Names a Big Five personality trait	100 percent	1 percent
Comments on nervousness, anxiety, or confidence	75 percent	0 percent
Numbered section format	99 percent	0 percent
Third person	100 percent	0 percent (second person)
Mean length	878 tokens	178 tokens

Table 6: Genre and content features of the reference summaries compared with system output.

A representative reference summary reads: "The most visible trait is Conscientiousness. The candidate's response is organized and logical...though she exhibits some signs of nervousness."

The reference summaries perform exactly the personality and emotional-state inference that this system's rubric and prompts are written to exclude. Reference-similarity metrics therefore measure, in part, willingness to make those inferences rather than evaluation quality alone.

4.8 Cost

System	Time per sample	Model calls per sample
Baseline	11.0 ± 7.2 s	1
Multi-agent (fixed order)	32.2 ± 11.6 s	4

Table 7: Cost per sample. Times are wall clock and dominated by API latency.

In orchestrator-routed mode the multi-agent pipeline makes approximately twelve model calls per sample, since each routing decision costs an additional call.

4.9 Evidence Span Localisation

An early configuration of the Evidence Agent produced spans averaging 55 seconds, frequently running from the first transcript segment to the last, which is technically accurate and explanatorily useless. Adding an explicit span-length constraint to the Evidence Agent prompt, capping spans at fifteen seconds and at most two adjacent transcript segments, reduced the mean span to 7.0 seconds with no loss of criterion coverage. The spans also distributed across the answer rather than all beginning at its start. This is a prompt-level change with the same model and the same sample, and it indicates that evidence localisation is controllable independently of scoring.

5 Discussion

The central hypothesis behind the multi-agent design was that separating evidence retrieval, rubric scoring, claim verification, feedback, and summarisation into distinct agents makes the system's failure modes visible rather than hidden inside one opaque call to a large multimodal model. The results support that hypothesis in a specific and limited sense.

The multi-agent pipeline produces outputs the baseline cannot: 4.72 evidence spans per sample, verification over every criterion, automatic capping of uncited scores, and a confidence value derived from evidence coverage. These are structural, not incremental. The worked example in Section 4.2 illustrates what this buys: the Verifier caught the pipeline's own Quality Agent assigning full marks for communication clarity on a basis the transcript could not settle, and the Summary Agent carried that uncertainty into the text the candidate reads. On the same sample, the baseline ended with a declaration of high confidence and had no mechanism that could have produced anything else.

The multi-agent pipeline did not, however, write summaries closer to the dataset's reference summaries, and the paired differences favour the baseline on all three judge dimensions at conventional significance levels. Two findings bear directly on how that result should be read. First, the hallucination gap is substantially an artifact of transparency: the multi-agent system reports its own scores and confidence, the reference summaries contain neither, and the judge counts the difference against it. On substantive claims about the candidate the two arms are indistinguishable. Second, the reference summaries themselves perform trait and emotional-state inference in every case, so a system that refuses to make those inferences is measured as less similar for reasons unrelated to evaluation quality.

The timestamp result is a genuine negative for one of the project's motivating assumptions. Given a transcript already annotated with timestamps in its prompt, the single-model baseline cited real timestamps at essentially the same rate as the agent pipeline. The advantage of the multi-agent design is therefore not that its citations are more likely to be real, but that they are attached to particular criteria in a structure that a downstream component can check, which is what makes verification possible at all.

Two implementation decisions proved more consequential than expected. Removing the synthesis agent in favour of deterministic assembly eliminated a redundant model call, removed an overlap with the Verifier, and converted confidence from a model assertion into a computed quantity. Separately, the span-length constraint in Section 4.9 reduced mean evidence span length by a factor of roughly eight without reducing coverage, which suggests that the practical quality of evidence grounding depends heavily on prompt-level constraints rather than on architecture alone.

Finally, the comparison in this report is between two arms running on the same model. A stronger model would be expected to improve both. The absolute judge ratings the multi-agent system achieved, above 4 out of 5 for helpfulness and faithfulness against a reference produced by a stronger model in a different format, are reasonable in that context, and the completeness rating of 3.43 is consistent with a 178-token summary being compared against an 878-token reference.

6 Limitations

- **The intended model was not evaluated.** All results use a small hosted model. The Qwen3-VL-8B configuration is implemented but was not run at benchmark scale, so the effect of model capacity on grounding quality is unmeasured.
- **No gold evidence spans exist.** RecruitView provides reference summaries but not annotated evidence timestamps, so temporal localisation accuracy, mean IoU and Recall at fixed thresh-

olds, could not be computed. Retrieval quality is therefore unquantified, and the localisation metrics implemented in the harness remain unexercised.

- **Reference summaries are model-generated.** All similarity metrics measure agreement with one model's output rather than correctness.
- **The judge shares a model family with the system under test**, which leaves open the possibility of self-preference bias.
- **CIDEr-D is degenerate at this length ratio** and is reported only with its length penalty disabled.
- **Visual evidence is retrieved but never verified.** Frame indices are produced by the Evidence Agent, but no component checks whether the model actually attended to those frames.
- **A single dataset, single language, and a dominant question type.** Generalisation across question formats, languages, and recording conditions is untested.
- **Feedback usefulness was never rated by humans**, only measured by proxy against reference summaries.
- **The completeness explanation is an inference.** The relationship between summary length and completeness rating was observed across two measurements but not tested directly.

7 Future Work

- **Evaluate on the intended open-weight model.** Running the benchmark on Qwen3-VL-8B served through vLLM would establish whether the grounding and verification behaviour observed here holds at higher model capacity, and would remove the dependency on a third-party API for data governance reasons.
- **Annotate gold evidence spans on a subset.** Hand-annotating evidence timestamps on twenty to thirty samples, with both annotators labelling an overlapping portion so that agreement can be reported, would complete the temporal localisation evaluation and turn retrieval quality from an unmeasured property into a reported one.
- **Verify visual evidence using attention.** An attention extraction module has been implemented against Qwen3-VL, loading the model through `transformers` with eager attention so that attention weights are returned, and rendering per-frame heatmaps over the cited frames. This would allow a mechanical check of whether the model attended to the frames it cited, extending verification from text into the visual modality. Following Jain and Wallace [9] and Wiegreffe and Pinter [10], any such result should be reported as showing where signal flowed rather than as establishing why a decision was made.
- **Reference summaries written under matched constraints.** Producing reference summaries that observe the same rubric and the same fairness constraints would remove the confound identified in Section 4.7 and allow similarity metrics to measure evaluation quality rather than willingness to infer traits.
- **Independent judging.** Using a judge from a different model family, or a panel of judges, would address the self-preference risk.
- **Human evaluation of feedback usefulness**, rated on relevance, specificity, actionability, clarity, and fairness by the people the feedback is written for.
- **Compare orchestration modes empirically.** Both the LLM-routed and fixed-order orchestration modes are implemented and produce identical report structures, but only the fixed-order

mode was benchmarked. Measuring whether learned routing produces different outcomes at roughly three times the call cost would resolve whether the orchestration agent earns its place.

- **Broaden the evaluation set** across question types, and extend to Arabic including code-switching, which would require replacing or extending the ASR and alignment components while leaving the architecture unchanged.

8 Conclusion

This project set out to build a trustworthy, explainable, multi-agent system for evaluating recorded interview answers, one that returns criterion-level scores tied to timestamped evidence, an explicit verification of every claim, and concrete feedback, instead of a single unexplained number.

The system is a complete implementation. A preprocessing pipeline turns a raw interview video into extracted audio, a word-aligned timestamped transcript, and labelled sampled frames. A model-agnostic provider abstraction supports self-hosted and API-hosted vision-language models behind one interface. Four agents, one deterministic assembly step, and a terminal summarisation agent, coordinated by an orchestrator over shared state, produce a structured report that a single-call baseline pipeline also emits, in the same schema, from the same inputs. A three-phase benchmark harness runs both arms, judges their output, and computes metrics offline.

Evaluated on 75 RecruitView samples, the system attaches timestamped evidence to 78 percent of rubric criteria, flags unsupported claims in 55 percent of samples, caps uncited scores automatically, and reports a confidence value computed from evidence coverage. It rated 4.60 out of 5 for helpfulness and 4.11 out of 5 for faithfulness against the dataset's reference summaries. The single-model baseline rated slightly higher on similarity and judge measures and ran roughly three times faster, though once the system's own reported scores and confidence values are excluded, both arms make claims about the candidate at statistically indistinguishable rates, and every reference summary in the dataset performs the personality inference this system is designed to exclude.

Evidence spans, verification verdicts, and a calibrated confidence signal are outputs the baseline does not produce at all. That is the measurable difference between the two architectures, and it is the property that makes an evaluation checkable by someone who was not present for it.

References

[1] AI4A-Lab. *RecruitView Dataset*. https://huggingface.co/datasets/AI4A-lab/RecruitView

[2] M. Bain. *WhisperX: Automatic Speech Recognition with Word-Level Timestamps*. https://github.com/m-bain/whisperX

[3] Qwen Team, Alibaba Group. *Qwen3-VL*. https://huggingface.co/Qwen

[4] W. Kwon et al. *Efficient Memory Management for Large Language Model Serving with PagedAttention*. Proceedings of the ACM SIGOPS 29th Symposium on Operating Systems Principles, 2023. https://github.com/vllm-project/vllm

[5] LangChain. *LangGraph: Workflows and Agents*. https://docs.langchain.com/oss/python/langgraph/workflows-agents

[6] *CrewAI Documentation*. https://docs.crewai.com/

[7] F. P.-W. Lo, J. Qiu, Z. Wang, H. Yu, Y. Chen, G. Zhang, and B. Lo. *AI Hiring with LLMs: A Context-Aware and Explainable Multi-Agent Framework for Resume Screening*. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops, 2025. https://arxiv.org/abs/2504.02870

[8] G. Sun, R. Yu, L. Yin, Y. Yang, B. Zhang, and Z. Xu. *CoMAI: A Collaborative Multi-Agent Framework for Robust and Equitable Interview Evaluation*. arXiv:2603.16215, 2026. https://arxiv.org/abs/2603.16215

[9] S. Jain and B. C. Wallace. *Attention is not Explanation*. Proceedings of NAACL-HLT, 2019.

[10] S. Wiegreffe and Y. Pinter. *Attention is not not Explanation*. Proceedings of EMNLP-IJCNLP, 2019.

[11] C.-Y. Lin. *ROUGE: A Package for Automatic Evaluation of Summaries*. Text Summarization Branches Out, ACL Workshop, 2004.

[12] R. Vedantam, C. L. Zitnick, and D. Parikh. *CIDEr: Consensus-based Image Description Evaluation*. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015.