

Maddies NLP at the FinEval 2026: Proposition-Aware Tagged Reasoning for Japanese Implicit Communication Recognition

Madhav Kapoor Aditya Channa Jastej Singh

Maharaja Agrasen Institute of Technology, Delhi, India

{madhav.kapoor13, adityachannadelhi, jastej28.singh}@gmail.com

Abstract

We describe our system for the FinNLP 2026 Japanese Financial Implicit Communication Recognition (JF-ICR) shared task. The task requires assigning one of five ordinal commitment and refusal labels to a Japanese investor-relations question and company response. Our final system uses the 30.7B-parameter Open-Router model `google/gemma-4-31b-it` in a two-stage, proposition-aware workflow. A label-blind first stage extracts tagged evidence for the target proposition, modality, actor and controllability, negation scope, conditionality, and institutional anchors. A second stage re-reads the original Japanese with this evidence and emits the final label before an optional rationale. On 253 labeled development examples, the system reaches 60.08% accuracy versus 56.92% for our earlier single-call control, with no invalid labels. The frozen submission scores 93.33% (14/15) publicly and 94.29% (33/35) privately. Deeper decision trees, latent-score mapping, and parallel judging all reduce accuracy, favoring compact decomposition with strict output control.

1 Introduction

Japanese investor-relations communication frequently conveys corporate intent through indirect modality, scoped negation, politeness, and conditional language. JF-ICR, introduced as part of the EBISU benchmark (Peng et al., 2026b), evaluates whether a company response signals strong or qualified commitment, neutrality, or strong or qualified refusal. The task is difficult because isolated lexical cues are unreliable. For example, a negative construction may reject a proposed action, deny the premise of the question, or commit to maintaining an existing policy.

Our system is based on a simple hypothesis: a language model should first identify proposition-aligned linguistic evidence and only then map that evidence onto the ordinal label set. We therefore

invoke the same eligible model (Gemma Team, 2026) twice. Stage 1 performs label-blind linguistic and pragmatic extraction. Stage 2 applies a fixed annotation rubric to the original Japanese and the extracted evidence. The intermediate representation is compact tagged text rather than JSON, which allows the second model call to consume useful evidence even when formatting is imperfect.

Our contributions are threefold:

- a proposition-aware, two-stage architecture that separates Japanese evidence extraction from ordinal adjudication;
- a label-first output contract and constrained repair path that yield 253/253 valid development predictions and 50/50 valid test predictions;
- an empirical comparison showing that several more complex routing and judging architectures underperform the compact two-stage system.

2 Task and Data

The shared task is part of the multilingual and multimodal financial evaluation setting described by Peng et al. (2026a). Each input contains a Japanese financial question and a company response. The output is one label from the five-point ordinal scale in Table 1. Ranking uses exact-label accuracy.

The released labeled set contains 253 examples: 29 labeled +2, 139 labeled +1, 72 labeled 0, 11 labeled -1, and 2 labeled -2. This distribution is strongly imbalanced toward qualified commitment. The participant test set contains 50 examples balanced across the five labels. The public leaderboard evaluates 15 examples (three per label), while 35 examples (seven per label) are reserved for the private leaderboard.

Label	Operational interpretation
+2	Strong, finalized commitment
+1	Qualified or conditional commitment
0	No sufficiently directional stance
-1	Qualified or temporary refusal
-2	Definitive refusal

Table 1: JF-ICR label semantics used by our adjudicator.

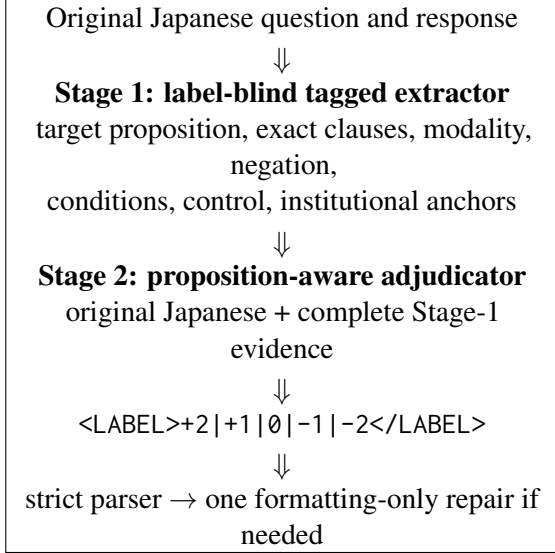


Figure 1: The selected two-stage tagged architecture.

We used only the organizer-provided question-response pairs and labels. We did not fine-tune the model, augment the training set, retrieve external demonstrations, or manually edit test predictions. Because prompt development involved inspection of public examples and their errors, the 253-example result is development evidence rather than a blind generalization estimate.

3 System Architecture

Figure 1 summarizes the final workflow. Both calls use google/gemma-4-31b-it through OpenRouter. It is a dense 30.7B-parameter model and satisfies the shared-task parameter limit.

3.1 Stage 1: Tagged Evidence Extraction

Stage 1 is explicitly prohibited from assigning a JF-ICR label. It reads the raw Japanese and produces at most three clause records plus a global summary. Each clause preserves an exact Japanese span and records the actor, company controllability, temporal orientation, speech act, polarity, negation scope, condition, final predicate, specificity, and institutional anchor. The global block

records whether a future stance exists, whether action is company-controlled, formal decision or implementation status, concrete targets, conditionality, answer coverage, non-disclosure scope, dominant clauses, and conflicting signals.

This schema was designed to capture errors that arise from proposition mismatch rather than from missing cue words. Strong evidence only matters when it scopes to the proposition in the question. Similarly, non-disclosure about details does not erase a substantive future stance elsewhere in the answer.

We use tagged text instead of strict JSON for inter-model communication. Japanese punctuation and long quoted spans require no escaping, and Stage 2 can interpret a partially imperfect field without a machine parser rejecting the entire analysis. Stage 1 is retried once only when the required outer analysis and global tags are absent.

3.2 Stage 2: Ordinal Adjudication

Stage 2 receives the original question, original response, and complete visible Stage-1 output. The prompt states that Stage 1 is evidence rather than authority and that the original Japanese wins in any conflict. The adjudicator follows a fixed sequence:

1. identify the exact target proposition;
2. determine whether the response expresses a relevant future stance;
3. resolve actor, controllability, negation scope, and direction;
4. separate action refusal from disclosure refusal or premise denial;
5. evaluate conditions, hedges, external dependence, finality, and institutional anchors;
6. resolve multiple clauses by proposition relevance and choose one label.

A central rule distinguishes continuity from refusal. “No intention to change policy X” may commit to maintaining X when the question asks about change, but may be a refusal when the question asks whether X will be introduced.

3.3 Output Control

The first output line must contain exactly one <LABEL> tag. An optional short rationale follows.

Architecture	Acc.	Macro-F1
Parallel ordinal judges	38.0	0.268
Latent scalar score	50.0	0.311
Single-call depth-4 tree	56.0	0.287
Multi-call decision tree	58.0	0.343
Dynamic boundary routing	62.0	0.404
Tagged label-first (ours)	68.0	0.386

Table 2: Paired architecture ablations on 50 reused development examples. Accuracy is reported as a percentage.

The parser accepts the strict tag and a small set of explicit near-miss markers; it never extracts an arbitrary numeric symbol from free-form reasoning. If parsing fails, only Stage 2 is called once more with a formatting-only repair instruction. Stage 1 is not regenerated, and unresolved failures stop submission creation rather than receiving a guessed fallback label.

For final inference we use temperature 0, top- p 1, seed 20260811, and four concurrent workers. Requests are cached by model, prompt, configuration, stage, and example. Providers were pinned through OpenRouter to DeepInfra Turbo and Friendli.

4 Development Experiments

4.1 Architecture Ablations

We evaluated architecture variants on the same 50-example development subset. This subset was reused across iterations, so Table 2 is a paired engineering comparison, not an independent estimate. The selected tagged system achieves the best accuracy. The dynamic boundary system slightly improves macro-F1, but loses three correct predictions, which is unfavorable under the task’s accuracy metric. Explicit decision trees overpredict rare negative labels, while parallel judges amplify disagreement instead of resolving it.

4.2 Full Development Set

Table 3 compares the selected system with our earlier frozen single-call control on all 253 labeled examples. Tagged decomposition improves accuracy by 3.16 points and produces no invalid predictions, although macro-F1 decreases because the rare refusal labels remain difficult. The paired comparison has $B = 28$ cases where the control is wrong and the tagged system is correct, versus $C = 20$ cases in the opposite direction. An exact paired binomial test gives a two-sided $p = 0.312$ (one-

System	Correct	Accuracy	Macro-F1
Single-call control	144/253	56.92	0.354
Tagged label-first	152/253	60.08	0.331

Table 3: Full labeled development-set results. Accuracy is a percentage.

Label	Support	Pred.	Precision	Recall
+2	29	25	0.400	0.345
+1	139	118	0.746	0.633
0	72	108	0.500	0.750
-1	11	2	0.000	0.000
-2	2	0	0.000	0.000

Table 4: Per-class development behavior of the selected system.

sided $p = 0.156$), so the development improvement is directional but not statistically conclusive.

4.3 Class Behavior

Table 4 shows that the development improvement is concentrated in the frequent commitment and neutral classes. The tagged system predicts +1 on 118 examples, compared with 103 for the earlier control and 139 in the gold data. Thus, the gain partly reflects correction of the control’s neutral bias, but the selected system still predicts +1 less often than its 54.9% development base rate. The largest remaining adjacent-boundary errors are 39 gold-+1 examples predicted as 0 and 15 gold-+2 examples predicted as +1.

The refusal classes remain unresolved on development: 10 of 11 gold -1 examples map to 0, and neither -2 example is recovered. With only 13 negative examples, architecture selection on refusal behavior would be unstable. We therefore retained the simpler system selected by overall accuracy and report the failure directly rather than introducing a separately tuned rare-class router.

4.4 Qualitative Error Analysis

Stage 1 is most useful when it preserves a future-oriented predicate together with its condition. In Japanese corporate answers, a caveat can occupy more surface text than the underlying commitment. A direct classifier may focus on uncertainty and choose 0, whereas the tagged representation keeps both the positive stance and its limitation visible to Stage 2. This is consistent with reduced neutral overprediction. Remaining errors require proposition-level distinctions: +2/+1 depends on finality rather than isolated formal vocabulary,

while negative decisions require separating refusal to act from refusal to disclose. Additional expert-labeled refusal examples would be more reliable than rules tuned to the 13 available cases.

5 Test Submission

The frozen final runner produced 50 valid predictions without retries or repairs. Its prediction counts were 11 for +2, 10 for +1, 12 for 0, 8 for -1, and 9 for -2, substantially closer to the known balanced test prior than the development distribution. It scored 93.33% (14/15) publicly and 94.29% (33/35) privately, or 94.00% (47/50) overall. The private leaderboard listed it fourth, tied on accuracy with two other submissions. We made no changes after either score.

The organizers released aggregate leaderboard scores but not instance-level test labels. We therefore cannot report a test confusion matrix or class-level recall, and cannot attribute the development-to-test gap to particular labels. The balanced test distribution removes the strong development prior, but greater item clarity is only a possible explanation for the gain rather than a verified finding.

6 Reproducibility and Efficiency

The submission is an API workflow using only google/gemma-4-31b-it through OpenRouter. Each example requires two normal calls, plus a formatting-only Stage-2 repair only when necessary. Final test inference used 100 calls, 84,779 prompt tokens, and 22,106 completion tokens. At the pinned providers' listed rates, the run cost was under USD 0.03.

The released runner validates that the input contains exactly IDs 253–302 and that the output contains exactly 50 rows with columns id, prediction. It rejects unknown labels and records raw outputs, cache keys, model configuration, prompt hashes, and source-data hashes. The frozen payload uses maximum outputs of 1,024 tokens for Stage 1, 256 for Stage 2, and 32 for a formatting repair. No API key is stored in the artifacts.

Code, prompts, configuration, and tests are publicly available in our public GitHub repository. The exact repository URL is <https://github.com/madhavkapoor13/fineval2026-jficr-review-source>. The manifest is written before calls and includes SHA-256 hashes of the input, runner, configuration, and all three prompts. This makes prompt

or routing changes visible during organizer verification. All model requests target the OpenRouter chat-completions endpoint; no direct provider API appears in the final workflow.

7 Limitations

Prompt and architecture design used the labeled public examples, so the 253-example results are susceptible to iterative development bias. Several ablations reuse the same 50 examples and should be interpreted only as paired regression tests. The 47/50 test result is also based on a small sample and remains uncertain. We did not conduct native-speaker error annotation for every model output. Finally, the two-call workflow doubles inference latency relative to direct classification, and API-provider behavior can change despite pinned model and provider names.

8 Ethics Statement

The task data consist of organizer-provided corporate and investor-relations question-response text. Our system does not infer personal attributes, use private external data, or provide investment recommendations. Intermediate model outputs may contain copied spans from the input and are sent to a third-party API provider, so deployment on confidential financial communication would require an appropriate data-processing agreement and retention controls. Labels summarize communicative stance and should not be interpreted as evidence of future corporate performance or legal commitment.

9 Conclusion

We presented a compact two-stage workflow for Japanese implicit communication recognition. Label-blind tagged extraction isolates proposition scope, modality, negation, and finality before ordinal classification, while a label-first contract makes the API pipeline reliable. The architecture improves full development accuracy over our earlier single-call control and reaches 47/50 across the complete balanced test set.

References

- Gemma Team. 2026. [Gemma 4 technical report](#). *arXiv preprint arXiv:2607.02770*.
- Xueqing Peng, Lingfei Qian, Yan Wang, Ruoyu Xiang, Yueru He, Yang Ren, Mingyang Jiang, Vincent Jim Zhang, Yuqing Guo, Jeff Zhao, et al. 2026a. [Multi-FinBen: Benchmarking large language models for multilingual and multimodal financial application](#). In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 16934–16963. Association for Computational Linguistics.
- Xueqing Peng, Ruoyu Xiang, Fan Zhang, Mingzi Song, Mingyang Jiang, Yan Wang, Lingfei Qian, Taiki Hara, Yuqing Guo, Jimin Huang, Junichi Tsujii, and Sophia Ananiadou. 2026b. [Ebisu: Benchmarking large language models in japanese finance](#). *arXiv preprint arXiv:2602.01479*.