GAMECOMMBENCH: A Unified Benchmark and Type-Aware Evaluation for AI-Generated Game Commentary

Anonymous ACL submission

Abstract

Game commentary is an open-ended generation task requiring multimodal perception, strategic reasoning, and contextual knowledge. Existing AI-Generated Game Commentary (AI-GGC) studies remain fragmented across games, modalities, and evaluation protocols, while overlap-based or holistic evaluators fail to capture the functional heterogeneity of commentary. We introduce GAMECOMMBENCH, a unified benchmark spanning board games, sports, and esports, with commentary aligned to heterogeneous game contexts and annotated by commentary type. We further propose Type-Aware Commentary Evaluation (TACE), a structured framework for evaluating different types of commentary. We then validate TACE for reliability and human agreement, and use it to benchmark representative AI commentators. Results reveal non-uniform capability profiles, with live observation and strategic analysis emerging as major bottlenecks. Together, GAMECOMMBENCH and TACE provide a diagnostic foundation for comparable and interpretable AI-GGC evaluation.

1 Introduction

Game commentary helps audiences make sense of gameplay in real time by narrating key events, interpreting important decisions, and situating moments in broader context (Behrens and Uhrich, 2022). AI-Generated Game Commentary (AI-GGC) has therefore become an increasingly active research area with clear practical value. Beyond utility, AI-GGC is a challenging open-ended NLG task: high-quality commentary requires live observation of unfolding events, strategic analysis of decisions and consequences, and historical recall of relevant game context (Zheng et al., 2025).

Despite growing interest, existing AI-GGC resources remain fragmented across individual games, including soccer (Giancola et al., 2018; Deliege et al., 2021), basketball (Yu et al., 2018; Yan et al., 2019), chess (Jhamtani et al., 2018; Lee et al., 2022), Go (Tomlin et al., 2022), and League of Legends(LoL) (Zhang et al., 2024; Wang and Yoshinaga, 2025). These resources have supported progress within their domains, but they remain tied to specific games, input modalities, data formats, and comment focuses. As a result, the field lacks a unified benchmark that organizes heterogeneous game contexts under a shared interface, making systematic cross-setting comparison difficult.

Evaluating game commentary poses an even deeper challenge. Many AI-GGC studies still rely on n-gram overlap metrics (Zheng et al., 2025), such as BLEU, ROUGE, or METEOR, despite the open-ended nature of commentary: for same game state, different commentators may produce distinct yet valid outputs. Such metrics therefore penalize variation rather than measuring commentary quality. Recent LLM-based evaluators move beyond lexical overlap (Liu et al., 2023), but existing approaches still score commentary holistically (Kim et al., 2025). This is insufficient because commentary serves distinct functions, including describing game states, interpreting strategic intent, and providing broader historical context (Zheng et al., 2025). Each function requires different criteria for judging quality, so evaluation should target these functions separately rather than collapse them into a single score. AI-GGC further complicates evaluation because a given game state admits neither a unique correct commentary nor a single correct reasoning process, especially for the analytical part.

To address these gaps, we present GAMECOMMBENCH and Type-Aware Commentary Evaluation (TACE). GAMECOMMBENCH brings board games, sports, and esports into a unified benchmark, while TACE evaluates different functional types of commentary with type-specific evaluation metrics. This design enables systematic comparison of AI com-

Code and Data are available here.

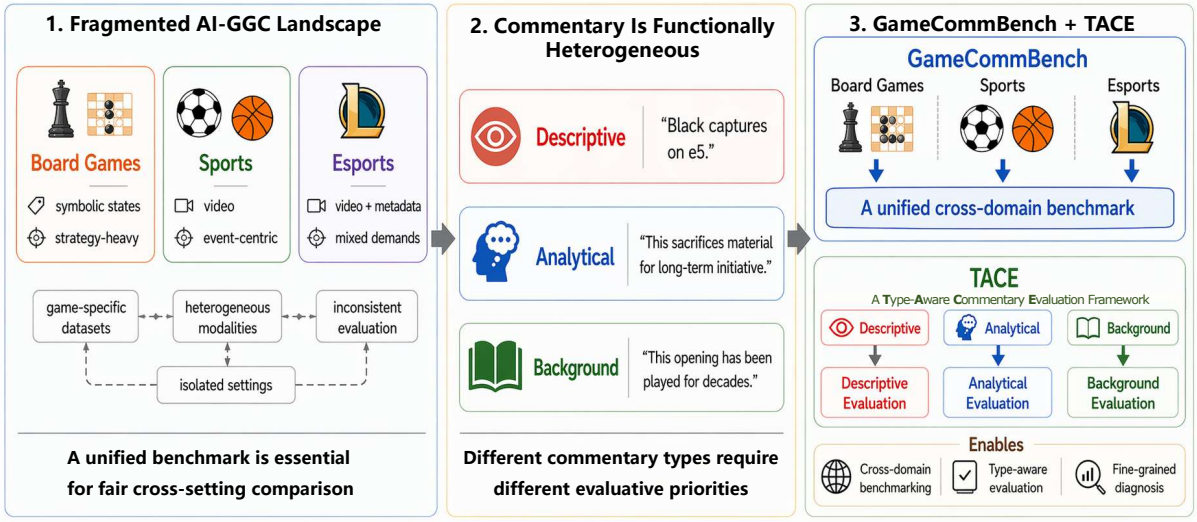

Figure 1: Existing AI-GGC research is fragmented across domains and modalities, and lacks effective evaluation schemes. We address these issues with GameCommBench, a unified cross-domain benchmark, and TACE, a commentary evaluation framework designed for descriptive, analytical, and background commentary.

mentators across games, and supports finer-grained diagnosis of model capabilities.

Our main contributions are as follows: (1) We construct GameCommBench, a unified AI-GGC benchmark with heterogeneous game contexts and type-annotated commentary. (2) We propose TACE that assesses descriptive, analytical, and background commentary with dedicated criteria instead of a single holistic score. (3) We validate TACE through repeated-run reliability analysis, human validation of LLM-mediated components, and cross-judge consistency checks. (4) We benchmark representative AI commentators and reveal non-uniform capability profiles, with live observation and strategic analysis remaining key bottlenecks.

2 Benchmark Construction

We construct GameCommBench as a unified AI-GGC benchmark spanning board games, sports, and esports. Figure 2 summarizes its detailed game coverage and data composition.

2.1 Benchmark Scope and Domain Coverage

GameCommBench covers five representative games across three domains: chess and Go for board games, soccer and basketball for sports, and LoL for esports. These games are selected not simply to broaden task coverage, but to expose variation in game-state representation, game dynamics, and demands on commentator abilities.

Board games provide relatively structured symbolic states but place strong demands on long-horizon strategic analysis (Lee et al., 2022). Commentary in board games typically emphasizes evaluating move quality, analysing strategy, and predicting possible future moves. Sports games, by contrast, are grounded in dynamic visual scenes, requiring commentators to identify salient events, track fast-changing actions, and produce coherent narration under temporal constraints (You et al., 2025). Esports introduce an additional layer of complexity, combining visually crowded, fast-changing gameplay with rich API-based information. This places higher demands on commentators to integrate complex multimodal inputs for game understanding and commentary generation (Zheng et al., 2025).

2.2 Data Collection and Processing Pipeline

Data Collection and Alignment. We collect aligned game contexts and commentary for the five games in GameCommBench from domain-specific sources. For chess and Go, we obtain game records and commentary from GameKnot and Foxwq, where comments are naturally aligned with turns. For basketball and soccer, we use broadcast videos and action labels from high-quality datasets, NSVA (Wu et al., 2022) for basketball and SoccerReplay-1988 (Rao et al., 2025) for soccer, and transcribe spoken commentary with WhisperX (Bain et al., 2023). For LoL, we collect official video clips paired with timestamped textual commentary; the timestamps are used to retrieve fine-grained API snapshots, yielding player-level and team-level game-state information. We also collect player and team background information from the same channels as supplementary context for commentary generation.

Data Cleaning and Quality Control. We combine manual screening and filtering with alignment-level repair. At the match level, we prioritize

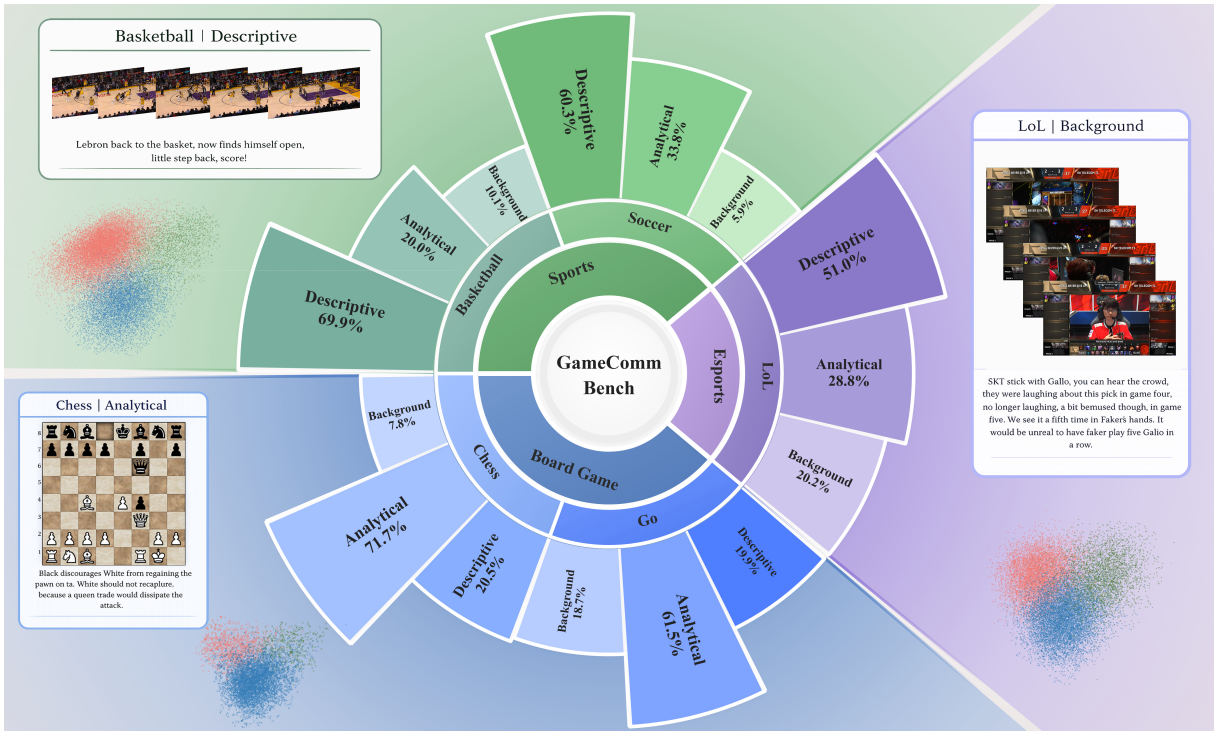

Figure 2: **Overview of GameCommBench.** The figure summarizes benchmark coverage, representative state-commentary examples, and commentary-type distributions across games. The central wheel reports the proportions of descriptive, analytical, and background commentary for each game. Scatter plots visualize classified human commentary embeddings, with red, blue, and green denoting descriptive, analytical, and background commentary.

games with dense commentary, competitive importance, and broad representativeness, while maintaining team and player diversity to avoid over-concentration on recurring entities. At the alignment level, board-game commentary naturally maps to discrete game states, whereas sports and esports rely on timestamp-based alignment and are more susceptible to sentence-boundary errors. To address this, we apply an LLM-based post-processing step that merges an incomplete first utterance in a segment with the previous segment. We further remove only content that is completely unrelated to the game itself, such as off-topic chatter or broadcast-side digressions. The original raw commentary is retained in each benchmark record for traceability and future research use.

Semantic Annotation. Since game commentary often mixes multiple functions, we decompose each commentary into finer semantic units and label them as descriptive, analytical, or background. This step is performed by LLM, and its agreement with human annotation is reported in Table 5. This unit-level labeling avoids forcing an entire utterance into a single type and allows TACE to evaluate each functional part separately.

Unified Packaging. Finally, all retained instances are packaged into a unified record schema. Each record contains raw and processed commentary aligned with game state, type-labeled semantic units, and game-related information. This schema keeps game-specific inputs such as symbolic states, videos, and API snapshots under a shared interface for generation and comparison.

2.3 Data Composition and Distribution

Figure 2 summarizes the benchmark composition across games. Since alignment units differ across domains, we report commentary-bearing turns for board games and aligned video clips for sports and esports. Specifically, the chess and Go subsets contain 2,159 and 2,857 turns, with 3,405 and 5,811 commentary sentences, respectively. In the sports domain, the soccer and basketball subsets contain 2,681 and 2,981 aligned video clips, paired with 5,893 and 6,120 sentences, respectively. For esports, the LoL subset provides 7,675 aligned clips and 10,001 sentences. Detailed commentary-type proportions for each game are shown in Figure 2.

Overall, GameCommBench exhibits substantial cross-game variation in scale, modality, and commentary composition. Board game commentary is more concentrated in analysing, sports commentary is predominantly descriptive, and esports commentary shows a more balanced distribution.

3 Type-Aware Commentary Evaluation

Game commentary is open-ended and contains different functional types, so a single holistic score is

insufficient. We propose TACE, which evaluates descriptive, analytical, and background commentary with criteria tailored to each type. Figure 3 summarizes the framework: commentary is decomposed by type and evaluated through three branches. We next describe the three pipelines and their metrics.

3.1 Descriptive Commentary Evaluation

Descriptive commentary focuses on *what is happening*. Therefore, instead of comparing surface text with human references, we measure whether the generated commentary accurately describes and covers salient events in the current game context.

For each instance, we construct a reference event set E_{ref} by extracting key actions from the descriptive units of the human commentary. To improve extraction accuracy, we incorporate domain-specific signals: dataset event labels for sports, explicit game-state representations for board games, and significant API changes for LoL. We use GPT-5.4 to extract reference events, GPT-5.4-mini to extract events from generated commentary, and GLM-4.7 to semantically align generated events with the reference event set. This extraction and matching pipeline is validated through human annotation in Section 4.2.

Given the aligned event sets, we define two complementary metrics. **Correctness**, computed as $|E_{\text{gen}} \cap E_{\text{ref}}|/|E_{\text{gen}}|$, measures the proportion of generated events supported by the reference set. **Completeness**, computed as $|E_{\text{gen}} \cap E_{\text{ref}}|/|E_{\text{ref}}|$, measures the fraction of reference events covered by the generated commentary. Correctness penalizes unsupported event descriptions, while completeness penalizes missing salient events.

3.2 Analytical Commentary Evaluation

Analytical commentary interprets move quality, state significance, or future trajectories, and its evaluation should prioritize the correctness of the reasoning rather than surface fluency. Open-ended reasoning evaluation is often simplified by turning the output into option selection, or by choosing tasks with a unique answer to limit open-endedness (Wen et al., 2025; Yao et al., 2025; Zhang et al., 2025). This setting does not fit AI-GGC: one game state may support multiple valid commentaries, and especially for analytical commentary, there is neither a single correct answer nor a single correct reasoning process. At the same time, evaluating lengthy analytical commentary holistically is also unreliable, since intermediate steps in a reasoning chain may be partially correct, unsupported, or internally inconsistent (Lightman et al., 2023).

We therefore need to verify whether each piece of articulated reasoning in the commentary is justified under the current game state. To achieve this, we decompose analytical text into granular, explicitly verifiable units. Specifically, each analytical commentary segment is converted into a directed graph, where nodes are *reasoning units*, the smallest logically meaningful propositions in the commentary, and edges denote predefined logical relations expressed in the text. We first decompose each analytical sentence into reasoning units, and then construct relations among them in two stages. **Intra-sentence relation modeling** considers candidate reasoning-unit pairs within the same sentence and determines whether an explicit logical relation is present. **Inter-sentence relation modeling** extends this process across sentences within the same analytical segment: we first retrieve a restricted set of candidate sentences using embedding and keyword similarity, and then construct relations across the paired sentences. Importantly, graph construction is purely text-based: no game-state or auxiliary information is provided. The resulting graph therefore represents the reasoning structure verbalized by the commentary itself. We then perform game-state-grounded verification on the constructed graph. At the node level, we assess whether each reasoning unit is supported by the corresponding game state. At the edge level, we assess whether the logical relation between two reasoning units is justified under the same game state. During verification, an edge is considered valid only when both endpoint reasoning units are supported and the expressed relation is justified. This decomposition mitigates the difficulty of LLM-as-a-Judge (Gu et al., 2024), enabling fine-grained assessment of strategic reasoning correctness instead of relying on holistic plausibility. Both relation modeling and graph verification are implemented using GPT-5.4-mini, and their accuracy is validated in Section 4.2.

Accordingly, we define two metrics for analytical commentary. **Analytical Faithfulness** measures the fraction of reasoning units and relations that pass game-state-grounded verification. **Reasoning Depth** measures the longest verified reasoning chain in the constructed graph, normalized by the largest value achieved by any evaluated model on the same instance. Together, they capture whether analytical reasoning is grounded and how far valid reasoning develops.

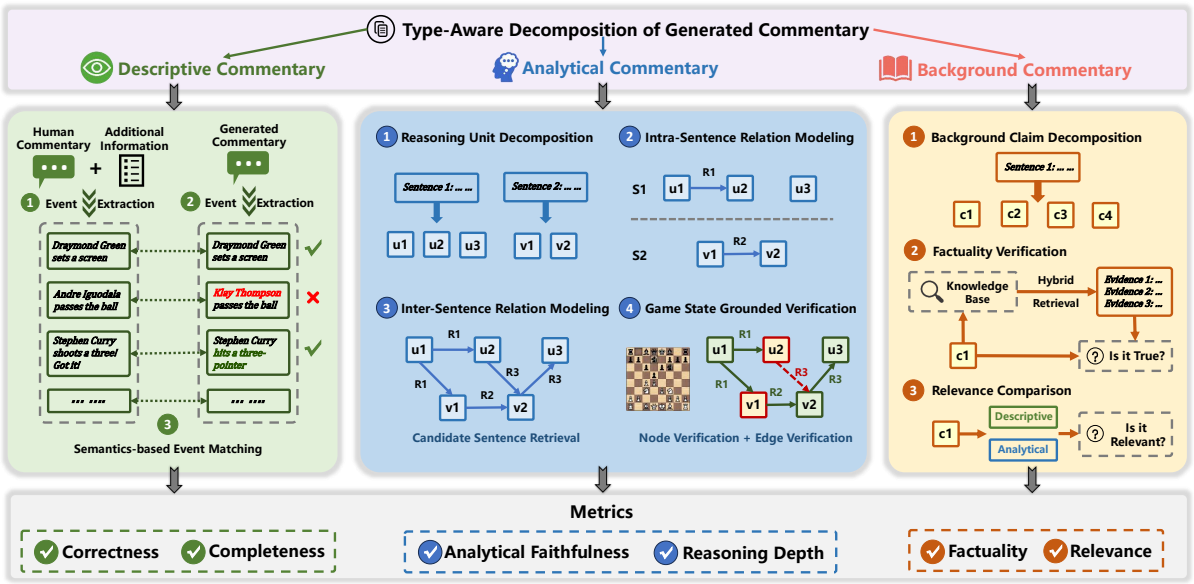

Figure 3: **Overview of TACE.** TACE evaluates descriptive, analytical, and background commentary through three type-specific pipelines. The resulting metrics assess the description correctness and completeness, reasoning faithfulness and depth, and background factuality and relevance.

3.3 Background Commentary Evaluation

Background commentary provides contextual or historical information beyond immediate game events. We evaluate it along two dimensions: **Factuality** and **Relevance**. For granular evaluation, we first decompose each background segment into *background claims*, defined as factual assertions distilled from the commentary.

Factuality assesses the objective accuracy of each background claim. Rather than relying solely on the evaluator's parametric knowledge, we support this judgment with an offline structured knowledge base constructed through iterative retrieval. This knowledge base provides external evidence for the relevant game, team, player, or match context. For each claim, we retrieve an evidence set using a hybrid strategy that combines indexed retrieval over structured fields, such as matches, teams, and players, with semantic retrieval based on claim–evidence similarity. The retrieved evidence is then provided to the evaluator for evidence-grounded factuality judgment, and the final factuality score is computed by averaging claim-level judgments over the background segment. **Relevance** evaluates whether a background claim aligns with the immediate context of the commentary. Specifically, we compare each claim against the game state and the remaining portions of the segment, including descriptive and analytical components, to determine whether the information is contextually integrated or unrelated. The relevance score is similarly derived by averaging claim-level judgments. Appendices C.4 and F.3.4 provide the detailed rubrics and evidence retrieval protocols.

4 Validation of TACE

We validate TACE along two axes: reliability and human agreement. Reliability measures run-to-run score variation and ranking consistency across repeated evaluations, while human validation checks whether the intermediate LLM-mediated decisions align with human annotations.

4.1 Reliability of TACE

To assess the reliability of TACE, we repeat the full evaluation pipeline three times on a held-out subset covering 6 models, 5 games, and 20 randomly sampled cases for each model-game pair. For each of the six TACE metrics, we report the mean and standard deviation across repeated runs, together with the coefficient of variation (CV) to quantify relative score fluctuation. Since benchmark conclusions also depend on model ordering, we further compute pairwise Spearman rank correlations across runs to assess ranking stability.

As shown in Table 1a, TACE exhibits low run-to-run variation across all six metrics. The CV values remain below 3.5%, indicating strong score-level reproducibility. Descriptive metrics show particularly small variation, while analytical metrics exhibit slightly higher but still limited fluctuation, as expected given the additional complexity of reasoning-graph construction and verification. Ranking consistency is also high, with Spearman correlations ranging from 0.856 to 0.965 across metrics. These results suggest that TACE provides stable type-specific evaluation signals for model comparison. We additionally report Spearman rank

Table 1: **Validation of TACE.** We report run-to-run reliability and human validation for LLM-mediated components.

(a) Reliability

Type	Metric	Mean $\pm$ SD	CV(%)	Spearman ρ
Desc.	Correctness	0.2312$\pm$0.0025	1.08	0.918
	Completeness	0.3215$\pm$0.0026	0.81	0.965
Analyt.	Faithfulness	0.3605$\pm$0.0112	3.11	0.905
	Reasoning Depth	0.0465$\pm$0.0016	3.44	0.856
Back.	Factuality	0.7358$\pm$0.0132	1.79	0.918
	Relevance	0.9185$\pm$0.0202	2.20	0.868

(b) Component accuracy

Type	Validated Step	Acc.(%)
Desc.	Event Extraction	91.16
	Event Matching	94.73
Analyt.	Relation Modeling	93.40
	Relation Verification	81.43
Back.	Factuality Judgment	92.86
	Relevance Judgment	94.44

correlations in Appendix E, where GPT-5.4-mini is replaced by Gemini-3-Flash as the judge model.

4.2 Human Validation

To assess the validity of TACE, we conduct human validation on the LLM-mediated components used in its evaluation pipeline. We first validate the accuracy of LLM-based type-aware decomposition. For each game, we randomly sample 100 decomposed commentary and manually label their types. The results, reported in Table 5, show consistently high decomposition accuracy across all games, with overall accuracy exceeding 95%.

Then, we validate whether each LLM-mediated step in TACE pipeline are consistent with human annotations. We use a stratified random sample covering all 6 evaluated models and 5 games, with three instances sampled for each model-game pair. This results in 90 annotated instances and more than 3,700 intermediate validation items, including event extraction outputs, event-matching decisions, reasoning-graph relations, verification judgments, and background-claim judgments. For descriptive commentary, we validate event extraction and semantic event matching. For analytical commentary, we validate whether the constructed relations accurately reflect the logical structure expressed in the text, and whether relation verification correctly judges the validity of reasoning connections with respect to the game state. For background commentary, we assess the accuracy of factuality and relevance judgments for extracted background claims. As shown in Table 1b, the LLM-mediated components achieve high agreement with human annotations across all components, with most validated steps exceeding 90% accuracy. These results support the validity of the intermediate automatic decisions used by TACE.

5 Experiments

Based on GameCommBench and TACE, we evaluate representative VLM commentators across games and commentary types, then analyze capability imbalances and the effects of model scale and reasoning mode.

5.1 Evaluation Setup

We evaluate six representative state-of-the-art VLMs: three closed-source models, Gemini-3.1-Pro-Preview, GPT-5.4, and Claude-Sonnet-4.5, and three open-weight models, Qwen3-VL-32B, GLM-4.6V, and InternVL3.5-38B. All models are evaluated on the full GameCommBench benchmark. Each instance includes basic player/team and game information. For board games, models receive symbolic game states with local move history, including the current move and the preceding four moves, corresponding to two full turns. For sports and esports, models process video inputs sampled at 4 FPS while preserving the original resolution; the prompt also specifies clip duration and asks the model to generate commentary appropriate for that duration. For esports, we additionally provide timestamp-aligned game-state metadata derived from the game API. Human commentary is not provided during generation.

All models use a shared commentary-generation template, with minor modality-specific adaptations for symbolic states, videos, and API-based metadata. We set temperature to 0 and the maximum output length to 1024 tokens, with one response generated per instance. To ensure comparable inference conditions, explicit reasoning modes are disabled when supported; otherwise, we use the minimal reasoning setting. Other decoding parameters are left at their default values.

We also evaluate two existing game-specific AI-GGC systems with TACE: Concept-guided Chess Commentary generation (CCC) (Kim et al., 2025) for chess and MatchVision (Rao et al., 2025) for soccer. Their results are included in the heatmaps in Figure 4, enabling comparison between general-purpose VLM commentators and task-specific AI-GGC systems.

Model	Descriptive		Analytical		Background	
	Correctness	Completeness	Faithfulness	Depth	Factuality	Relevance
Gemini-3.1-Pro-Preview	0.206	0.499	0.258	0.171	0.691	0.865
GPT-5.4	**0.281**	**0.610**	**0.439**	**0.318**	0.655	0.922
Claude-Sonnet-4.5	0.243	0.533	0.300	0.196	0.661	0.911
Qwen3-VL-32B	0.217	0.467	0.163	0.197	0.671	**0.927**
GLM-4.6V	0.220	0.282	0.190	0.185	0.682	**0.927**
InternVL3.5-38B	0.196	0.445	0.124	0.114	**0.734**	0.866
Average	0.227	0.473	0.246	0.196	0.683	0.903

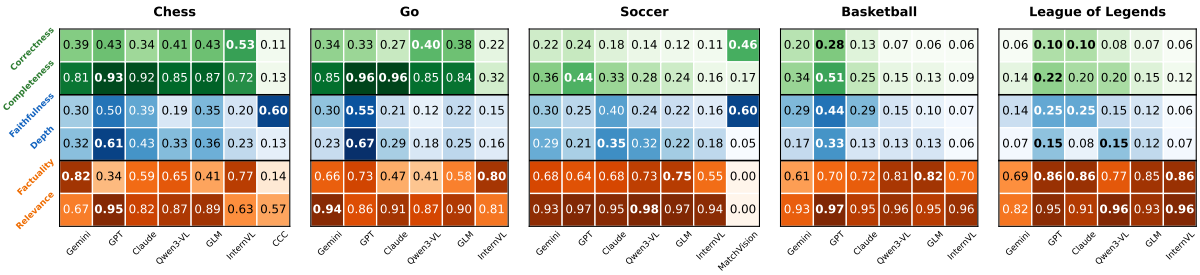

Figure 4: **Main benchmark results on GameCommBench.** The top table reports aggregated model scores for the six TACE metrics, with the average row summarizing metric-wise performance across evaluated models. The bottom heatmaps show game-wise capability profiles across models and metrics, revealing cross-game and type-specific performance patterns.

5.2 Main Benchmark Results

Figure 4 summarizes the overall and game-wise performance of the six VLM commentators. The results reveal substantial capability imbalance across both games and commentary types. No model consistently dominates all metrics: GPT-5.4 performs best on descriptive and analytical metrics, whereas other models remain competitive on background factuality or relevance. This confirms that AI-GGC capability is not one-dimensional and should not be collapsed into a single holistic score.

For descriptive commentary, models show a clear gap between completeness and correctness. The average completeness reaches 0.473, while correctness is only 0.227, suggesting that VLMs often generate more event descriptions than can be directly supported by the reference signals. This pattern indicates a verbosity–grounding mismatch: models often describe plausible events, but many of these descriptions are not supported by the reference events or available game-state signals. In board games, this issue is partly mitigated by the compact symbolic input, and lower correctness may also reflect the fact that human commentators do not mention every objectively valid move-level event. However, the same pattern becomes much more problematic in video-based games. For sports and esports, both correctness and completeness drop sharply, as shown in Figure 4, indicating that current VLMs not only miss salient events but also generate unsupported descriptions in fast-paced multi-agent visual scenes. This exposes live observation as a major bottleneck for AI-GGC in dynamic video settings.

Analytical commentary remains a major bottleneck. The average analytical faithfulness is 0.246 and the average reasoning depth is 0.196, showing that most models have difficulty producing game-grounded strategic reasoning. GPT-5.4 achieves the best analytical performance, but the absolute scores remain limited. This indicates that even strong VLMs often produce analyses that are shallow, partially unsupported, or weakly connected to underlying game state. For reference, human analytical commentary shows high faithfulness (0.78) but relatively low depth (0.28), suggesting that expert commentary can be strategically correct while remaining concise, rather than explicitly spelling out long reasoning chains. This distinction also shows why faithfulness and depth should be interpreted as complementary analytical signals.

Background commentary shows a different capability profile. Models achieve high relevance, reaching 0.903 on average, while factuality remains moderate at 0.683. Notably, open-weight models achieve the strongest scores on both background metrics, unlike the pattern observed for descriptive and analytical metrics. The two AI-GGC systems show limited performance on background metrics, indicating that historical recall remains underexplored in existing AI-GGC work. Overall, current VLMs can usually generate background information that is topically connected to the commentary

Table 2: **Effects of model scale and reasoning mode.** Left: comparison of open-weight models with different sizes. Right: comparison between minimal-reasoning and thinking settings on board games.

(a) Effect of Model Scale

Model	Size	Descriptive		Analytical		Background	
		Correctness	Completeness	Faithfulness	Depth	Factuality	Relevance
Qwen3-VL	32B	0.252	0.533	0.169	0.232	0.647	0.920
	8B	0.260	0.500	0.111	0.184	0.566	0.916
InternVL3.5	38B	0.229	0.322	0.148	0.151	0.704	0.835
	8B	0.216	0.319	0.137	0.188	0.580	0.887
GLM-4.6V	106B	0.247	0.520	0.227	0.231	0.639	0.926
	9B	0.257	0.492	0.160	0.112	0.511	0.923

(b) Effect of Reasoning Mode

Model	Thinking Level	Analytical	
		Faithfulness	Depth
Gemini	Min	0.206	0.271
	Max	0.532	0.451
GPT	Min	0.250	0.427
	Max	0.364	0.445
Claude	Min	0.235	0.394
	Max	0.246	0.357

context, likely benefiting from parametric knowledge and the basic game information provided in the prompt. However, there is still room for improvement for the factuality, suggesting that background commentary may benefit from retrieval-augmented or evidence-grounded generation mechanisms that provide verified player, team, match, or historical information.

5.3 Scale and Reasoning Mode

We further analyze two factors that may affect commentary capability: model scale and reasoning mode. For model scale, we compare the three open-weight models in the main result with their lighter variants on the full board-game and sports subsets. For reasoning mode, we evaluate the three closed-source models on a randomly sampled quarter of the board-game turns, comparing the minimal-reasoning setting used in the main experiments with the maximum thinking setting supported by each model.

As shown in Table 2a, increasing model scale does not lead to uniform improvements in descriptive commentary. Larger models generally achieve slightly higher completeness, but their correctness remains similar or even decreases. This suggests that scaling makes models describe more events, but not a higher proportion of supported events. This pattern is consistent with the main benchmark results, where descriptive correctness remains a key bottleneck, especially for video-based games.

The effect of scale is more evident for analytical and background capabilities. Larger models generally improve analytical faithfulness, indicating better grounding of strategic statements in the game state. However, reasoning depth does not increase consistently across model families, suggesting that scale alone does not guarantee longer valid reasoning chains. For background commentary, factuality shows the clearest scaling trend, with larger variants consistently achieving higher factuality scores. This may reflect stronger parametric knowledge, better entity-level understanding, or more effective use of the basic game information provided in the prompt. In contrast, relevance is already high for most models and is less sensitive to scale.

As shown in Table 2b, reasoning mode has a clear positive effect on analytical commentary. Across the three closed-source models, increasing the thinking level improves five out of six analytical scores. The gains are particularly strong for Gemini-3.1-Pro-Preview, whose faithfulness and reasoning depth both increase substantially, making it the strongest model in this board-game analytical setting. GPT-5.4 also benefits from the higher thinking setting, with a clear improvement in faithfulness and a smaller gain in reasoning depth. Claude-Sonnet-4.5 shows a weaker response: faithfulness slightly improves, while reasoning depth decreases modestly. Overall, these results indicate that stronger thinking settings can improve analytical commentary, especially analytical faithfulness, although their effect on reasoning depth remains more model-dependent.

6 Conclusion

We introduced GAMECOMMBENCH, a unified benchmark for AI-GGC across board games, sports, and esports, together with TACE, a type-aware commentary evaluation framework for descriptive, analytical, and background commentary. Our validation shows that TACE provides stable and human-aligned evaluation signals. Experiments with representative AI commentators reveal non-uniform capability profiles across games and commentary types, with grounded game state description and faithful strategic reasoning remaining key bottlenecks. Further analyses show that model scale and reasoning mode bring uneven gains across commentary capabilities. Together, GAMECOMMBENCH and TACE support more comparable and interpretable evaluation of AI-GGC.

Limitations

GAMECOMMBENCH is designed to cover representative AI-GGC settings across board games, sports, and esports, but it does not exhaust all game genres, languages, or broadcasting styles. The current benchmark focuses on English text commentary generation and evaluation over five games with different state representations; it does not address speech-based commentary, voice delivery, or other multimodal presentation forms of game broadcasting. Extensions to additional languages, regional commentary styles, game communities, and speech or multimodal commentary settings may require new data collection and domain-specific processing. Finally, full multimodal evaluation across multiple games and VLMs is costly, especially with closed-source APIs.

Ethical Considerations

This work studies benchmark construction and evaluation for AI-GGC. Generated commentary may contain errors, such as inaccurate descriptions or unsupported interpretations, which could mislead viewers if used directly in public-facing commentary systems, especially in text-only live updates where viewers rely primarily on the generated text. In addition, GAMECOMMBENCH is constructed from game records, videos, transcripts, API-derived metadata, and related contextual information. These sources may contain public names of players, teams, matches, or commentators that are necessary for game commentary and background evaluation; however, we do not collect private or sensitive personal information or intentionally include offensive content. The benchmark is constructed from public or previously released game materials, and no separate consent was obtained from players, teams, or commentators. Users of the benchmark should respect the licenses, terms of use, and attribution requirements of the original data sources.

References

Max Bain, Jaesung Huh, Tengda Han, and Andrew Zisserman. 2023. Whisperx: Time-accurate speech transcription of long-form audio. *arXiv preprint arXiv:2303.00747*.

Anton Behrens and Sebastian Uhrich. 2022. You'll never want to watch alone: The effect of displaying in-stadium social atmospherics on media consumers' responses to new sport leagues across different types of media. *European Sport Management Quarterly*, 22(1):120–138.

Adrien Deliege, Anthony Cioppa, Silvio Giancola, Meisam J Seikavandi, Jacob V Dueholm, Kamal Nasrollahi, Bernard Ghanem, Thomas B Moeslund, and Marc Van Droogenbroeck. 2021. Soccernet-v2: A dataset and benchmarks for holistic understanding of broadcast soccer videos. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4508–4519.

Silvio Giancola, Mohieddine Amine, Tarek Dghaily, and Bernard Ghanem. 2018. Soccernet: A scalable dataset for action spotting in soccer videos. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 1711–1721.

Jiawei Gu, Xuhui Jiang, Zhichao Shi, Hexiang Tan, Xuehao Zhai, Chengjin Xu, Wei Li, Yinghan Shen, Shengjie Ma, Honghao Liu, and 1 others. 2024. A survey on llm-as-a-judge. *The Innovation*.

Harsh Jhamtani, Varun Gangal, Eduard Hovy, Graham Neubig, and Taylor Berg-Kirkpatrick. 2018. Learning to generate move-by-move commentary for chess games from large-scale social forum data. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1661–1671.

Jaechang Kim, Jinmin Goh, Inseok Hwang, Jaewoong Cho, and Jungseul Ok. 2025. Bridging the gap between expert and language models: Concept-guided chess commentary generation and evaluation. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 9497–9516.

Andrew Lee, David Wu, Emily Dinan, and Mike Lewis. 2022. Improving chess commentaries by combining language models with symbolic reasoning engines. *arXiv preprint arXiv:2212.08195*.

Hunter Lightman, Vineet Kosaraju, Yuri Burda, Harrison Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. 2023. Let's verify step by step. In *The twelfth international conference on learning representations*.

Yang Liu, Dan Iter, Yichong Xu, Shuohang Wang, Ruochen Xu, and Chenguang Zhu. 2023. G-eval: Nlg evaluation using gpt-4 with better human alignment. In *Proceedings of the 2023 conference on empirical methods in natural language processing*, pages 2511–2522.

Jiayuan Rao, Haoning Wu, Hao Jiang, Ya Zhang, Yanfeng Wang, and Weidi Xie. 2025. Towards universal soccer video understanding. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 8384–8394.

Nicholas Tomlin, Andre He, and Dan Klein. 2022. Understanding game-playing agents with natural language annotations. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 797–807.

Zihan Wang and Naoki Yoshinaga. 2025. Commentary generation from multimodal game data for esports moments in multiplayer strategy games. In *Proceedings of the 14th International Joint Conference on Natural Language Processing and the 4th Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics*, pages 1795–1807.

Qianfeng Wen, Zhenwei Tang, and Ashton Anderson. 2025. Chessqa: Evaluating large language models for chess understanding. *arXiv preprint arXiv:2510.23948*.

Dekun Wu, He Zhao, Xingce Bao, and Richard P Wildes. 2022. Sports video analysis on large-scale data. In *European conference on computer vision*, pages 19–36. Springer.

Yichao Yan, Ning Zhuang, Bingbing Ni, Jian Zhang, Minghao Xu, Qiang Zhang, Zheng Zhang, Shuo Cheng, Qi Tian, Yi Xu, and 1 others. 2019. Fine-grained video captioning via graph-based multi-granularity interaction learning. *IEEE transactions on pattern analysis and machine intelligence*, 44(2):666–683.

Huanjin Yao, Jiaxing Huang, Yawen Qiu, Michael K Chen, Wenzheng Liu, Wei Zhang, Wenjie Zeng, Xikun Zhang, Jingyi Zhang, Yuxin Song, and 1 others. 2025. Mmreason: An open-ended multi-modal multi-step reasoning benchmark for mllms toward agi. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 273–283.

Ling You, Wenxuan Huang, Xinni Xie, Xiangyi Wei, Bangyan Li, Shaohui Lin, Yang Li, and Changbo Wang. 2025. Timesoccer: An end-to-end multimodal large language model for soccer commentary generation. In *Proceedings of the 33rd ACM International Conference on Multimedia*, pages 3418–3427.

Huanyu Yu, Shuo Cheng, Bingbing Ni, Minsi Wang, Jian Zhang, and Xiaokang Yang. 2018. Fine-grained video captioning for sports narrative. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6006–6015.

Yiran Zhang, Mo Wang, Xiaoyang Li, Kaixuan Ren, Chencheng Zhu, and Usman Naseem. 2025. Turnbench-ms: A benchmark for evaluating multi-turn, multi-step reasoning in large language models. *Findings of the Association for Computational Linguistics: EMNLP*, pages 19892–19924.

Zhihao Zhang, Feiqi Cao, Yingbin Mo, Yiran Zhang, Josiah Poon, and Caren Han. 2024. Game-mug: Multimodal oriented game situation understanding and commentary generation dataset. *arXiv preprint arXiv:2404.19175*.

Qirui Zheng, Xingbo Wang, Keyuan Cheng, Muhammad Asif Ali, Yunlong Lu, and Wenxin Li. 2025. From multimodal perception to strategic reasoning: A survey on ai-generated game commentary. *arXiv preprint arXiv:2506.17294*.

A Detail of Data Collection and Processing Pipeline

This appendix provides implementation-level details for the data collection and processing pipeline among the five game genres in GAMECOMMBENCH. As shown in Figure 5, the main text already describes the high-level collection, cleaning, and packaging strategy; here we focus on the concrete processing steps used to convert raw game state data and commentary into cleaned benchmark instances.

A.1 Unified record schema and normalization

The final conversion stage packages heterogeneous samples into a common JSONL interface. Sports records use the complete schema below, and board-game/esports records follow the same semantic contract while using PGN, SGF, or API-derived fields as the game-state carrier.

```
{
  "No": "...",
  "game_state": {
    "modality": "video | pgn | sgf | api+video",
    "storage_file": "...",
    "step": "...",
    "label": "..."
  },
  "background": "...",
  "commentary": {
    "raw_commetary": "...",
    "processed_commentary": "...",
    "commentary_category": [
      {
        "No": "...",
        "content": "...",
        "category": "descriptive | analytical |
        background"
      }
    ]
  }
}
```

Here, `raw_commetary` keeps the field name used by the implementation, and category IDs are normalized into `descriptive`, `analytical`, and `background`. For football and basketball, the original video or event ID is replaced by an MD5-based stable identifier, with the original-to-hash mapping retained separately.

A.2 Game-specific processing workflows

This subsection merges the collection, alignment, cleaning, and final annotation steps for each game

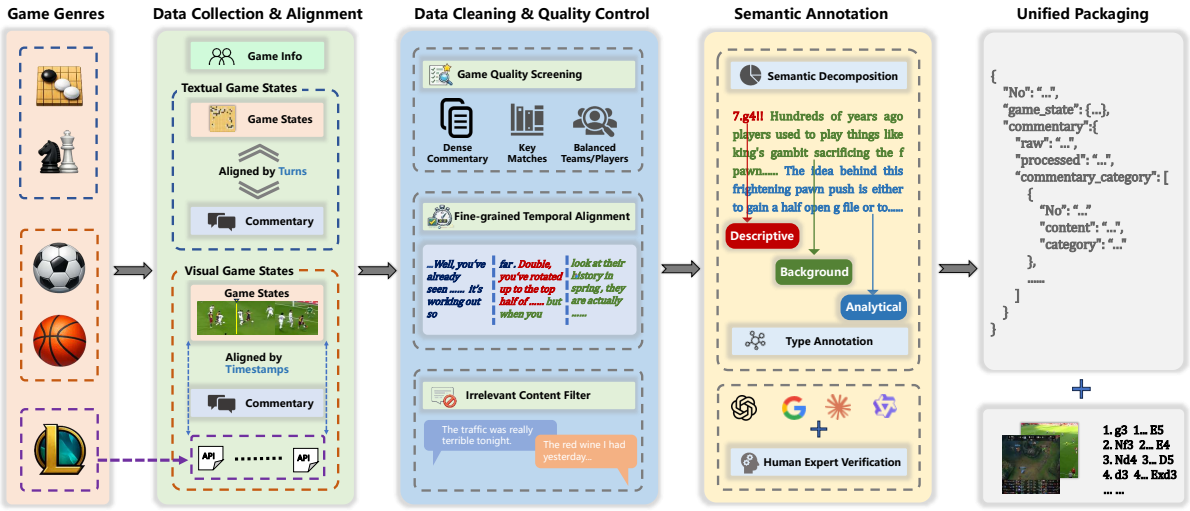

Figure 5: **Detailed data processing pipeline.** Raw game data are collected and aligned, cleaned through quality control and temporal repair, decomposed into commentary types, and packaged into a unified format.

type. The degree of processing differs by source modality: chess is the lightest because the source comments are already attached to PGN plies; Go requires extraction from FoxWQ records plus removal of unusable commentary; football, basketball, and LoL require explicit temporal alignment and cleaning. The LLM prompts used by the released processing code are reproduced after the workflow descriptions.

A.2.1 Football

Football starts from raw rows containing a clip caption and ASR-style commentary. The retained rows are tied to externally supplied video clips through the football video manifest, and match-level background is reconstructed from the raw match metadata, including teams, date, venue, referee, and player lists. The processing chain then performs: (i) sentence segmentation and type classification over the transcript with Prompt A.1; (ii) caption-grounded filtering with Prompt A.3, where the caption is treated as the authoritative action description and sentences with ambiguous references, weak relevance, or contradictions are removed; (iii) caption-commentary fusion with Prompt A.4, which creates a unified broadcast-style description from the caption and retained commentary; (iv) a final pass of Prompt A.1 over the fused text; and (v) conversion into the unified schema, including label normalization and MD5-based stable identifiers.

A.2.2 Basketball

Basketball follows the same alignment-and-cleaning pattern as football, but its initial alignment is stricter because the raw NSVA-derived caption and the transcript may refer to adjacent plays. The pipeline first applies the relevance gate in Prompt A.5 and only writes caption-commentary pairs whose boxed judgment is `\boxed{true}`. The remaining rows are segmented and classified with the basketball version of the classification prompt in Prompt A.2, filtered with the same retention logic as Prompt A.3 using the basketball-specific phrase "on-court situations," and then fused with the basketball caption-commentary prompt in Prompt A.6. The basketball pipeline additionally runs the quality selector in Prompt A.7 over batches of three merged candidates and keeps the two highest-ranked examples. Final conversion attaches the raw NSVA caption and transcript, the video storage path, the coarse action label, the game background recovered from NBA metadata files, normalized commentary categories, and an MD5-based stable identifier.

A.2.3 Chess

Chess uses the lightest data-processing path. Saved GameKnot HTML pages are grouped by game ID, sorted by page index, and parsed into player/event metadata plus move-comment tuples. The converter writes one PGN file per game, preserves comments in PGN braces, removes HTML/script artifacts from comments, normalizes whitespace, escapes PGN tag strings, and stores metadata such as event, site, players, ratings, opening, annotator, game ID, and move count. Since commentary is already attached to moves, no timestamp matching or caption-grounded LLM cleaning is needed. The

Game	Primary alignment	Processing summary
Football	Caption/event clip to transcript	Segment and classify transcript, filter against caption, fuse caption and retained commentary, reclassify, then normalize IDs and labels.
Basketball	NSVA caption/action clip to transcript	Gate caption-commentary relevance, segment/classify, filter, fuse, select high-quality merged outputs, then normalize with NBA metadata and action labels.
Chess	PGN mainline ply	Parse GameKnot HTML into PGN comments; preserve move-attached commentary with only lightweight text normalization.
Go	SGF mainline ply	Crawl FoxWQ JSON, extract SGF, discard empty or irrelevant comments, and normalize turn-aligned records.
League of Legends	Subtitle window plus API window	Align subtitle segments to clips, aggregate Riot/LoL Esports API windows, and retain significant team/player state changes.

Table 3: Integrated data-processing workflows for the five game types.

final records use the PGN file as `storage_file`, the ply as `step`, and the comment text as the reference commentary after lightweight normalization and label mapping.

A.2.4 Go

Go starts from FoxWQ metadata seeds. The crawler downloads each referenced JSON payload, and the converter extracts the `chess` field into an SGF file named by `chessid`. Empty or malformed JSON entries are skipped. Compared with chess, Go needs an additional cleanup step because extracted records can contain empty, duplicated, or off-topic comments; these are removed before the retained move-comment pairs are normalized into the released schema. SGF coordinates are kept as the source representation, while later state construction can convert them into standard Go coordinates and render the board before the target move.

A.2.5 League of Legends

LoL combines subtitle/VOD alignment with API-derived state extraction. Subtitle files are parsed into timestamped segments after HTML tag removal, whitespace normalization, and optional consecutive-duplicate removal. For Match-V5 data, events are aggregated in fixed time windows, the most-overlapping subtitle sentence is selected as reference commentary, and each row stores the event window, linearized event summaries, team K/D/A and gold, patch, game date, and a stable MD5 row ID. For LoL Esports live-stat data, the acquisition script resolves match and game IDs from team names, game numbers, league schedules, and event details, fetches live-stat windows, and extracts team/player snapshots around the target timestamp. When start/end snapshots are available, significant API changes are retained as structured context through `step_boundary_change`; this records boundary-level changes such as team gold, team kills, and player-level metric changes.

B Candidate Commentary Generation

Candidate commentaries are generated without access to the human reference commentary. Each generator reads one JSONL record, constructs the game-specific input context, calls the target model endpoint, and writes a JSONL output with `No` and `commentary_by_{model}`.

For football and basketball, videos are sampled at 4 FPS and the sampled frames are interleaved with timestamp text while preserving the original frame resolution used in the run. For chess and Go, the scripts parse the source game files and pass symbolic state rather than visual frames: chess reconstructs FEN before/after the target move together with SAN/UCI notation and recent move history, while Go renders the board before the target move and supplies the current coordinate plus recent moves. League of Legends uses video input together with match background and, when available, the `step_boundary_change` API summary describing team/player state transitions. Across all reported generation runs, temperature is set to 0 and reasoning or thinking modes are disabled when supported.

The prompts below are copied from the candidate-generation scripts. Runtime placeholders are filled by the corresponding script immediately before the model call.

C Evaluation

The evaluation code implements TACE as a namespaced pipeline with four functional branches: segmentation, descriptive evaluation, analytical evaluation, and background evaluation. The run scope can be set to `all`, `segmentation`, `descriptive`,

Game	Input supplied to model	Decoding policy	State construction
Football	Timestamped video frames, match background, clip duration	temperature 0; thinking disabled	4 FPS visual sampling with original frame resolution.
Basketball	Timestamped video frames, match background, clip duration	temperature 0; thinking disabled	4 FPS visual sampling with original frame resolution.
Chess	FEN before/after, SAN, UCI, move string, recent moves, background	temperature 0; thinking disabled	PGN mainline parsing with recent move history.
Go	ASCII board, SGF and standard coordinate, recent moves, background	temperature 0; thinking disabled	SGF mainline parsing with board rendering before the current move.
League of Legends	Video input, match background, API transition context	temperature 0; thinking disabled	Clip-aligned visual input plus team/player state transitions.

Table 4: Candidate commentary generation settings used by the provided scripts.

`analytical`, or `background`. Judge and extraction calls use temperature 0.0, and sport aliases are normalized before evaluation so that `football`, `soccer`, and their Chinese alias map to the same branch.

C.1 Segmentation for evaluation

Before type-specific scoring, candidate commentary is segmented into the smallest independent semantic units and assigned category IDs 1, 2, or 3. The implementation validates the returned JSON array and then routes units with ID 1 to descriptive evaluation, ID 2 to analytical evaluation, and ID 3 to background evaluation. The football prompt is shown below as the concrete representative; all other sports use parallel prompt files with the same output contract.

C.2 Descriptive evaluation

Descriptive evaluation measures whether the generated commentary correctly captures the salient observable events. The evaluator first obtains a non-empty reference event list from the human reference and state context. It then extracts generated events from candidate descriptive units. For sports, the reference extractor uses labels and match background; for chess and Go, it uses the symbolic board context; for League of Legends, it uses both commentary and API-derived state changes. The same reference event list is used for both completeness and correctness matching. If no descriptive candidate event is extracted, both metrics are set to zero. Scores are rounded to two decimals in the implementation.

C.3 Analytical evaluation

Analytical commentary is evaluated through a proposition graph rather than a holistic score. The pipeline first extracts explicit analytical propositions from category-2 units. It then builds sparse forward edge candidates using adjacent propositions and hybrid retrieval, where similarity combines embedding similarity, entity overlap, and keyword overlap. Candidate edges are restricted to the relation set `CAUSE`, `REASON`, `CONDITION`, `CONTRAST`, and `ELABORATION`. A second audit stage removes redundant or weakly anchored edges. The resulting graph is validated as a DAG with no self-loop and no duplicate edge for the same source-target pair.

The released implementation stores several analytical diagnostics. The TACE faithfulness signal is the graph-grounded correctness ratio, computed from node and final-edge truth labels:

$$\text{AnalyticalFaithfulness} = \frac{N_{\text{true}} + E_{\text{true}}}{N + E}, \quad (1)$$

where N and E are the number of judged propositions and graph edges. The code also reports `proposition_truth`, which is the node-only truth ratio. Reasoning depth is reported through `effective_depth`: after node and edge verification, the evaluator keeps only true nodes and true edges, computes the longest valid unweighted DAG path, and normalizes it by the maximum raw depth among models for the same sport-record group. Raw graph `depth` and `relation_info` are retained as additional diagnostics.

C.4 Background evaluation

Background evaluation first converts category-3 units into verifiable claims. Each claim carries a source sentence ID, claim type, focus key, entities, and time scope, making the later factuality and relevance judgments operate on independently checkable units.

Factuality is judged against a local background knowledge base derived from the reference record metadata. The retriever combines structured chan-

Table 5: Human validation accuracy of LLM-based type-aware decomposition across games.

Game	Accuracy
Chess	94.0%
Go	93.0%
Football	98.0%
Basketball	98.0%
LoL	96.0%
Overall	95.8%

nels based on source ID, focus key, entity alignment, fact-key alignment, and source-context fallback with optional semantic embedding retrieval. The evidence package is bounded by a hard evidence-overflow limit, and the factuality judge may only cite evidence IDs that were actually retrieved. If no valid evidence is available, the claim is labeled false by construction. Relevance is judged separately by comparing each claim with the immediate descriptive and analytical clauses of the same candidate commentary. The reported scores:

$$\text{Factuality} = \frac{\#\text{claims labeled true}}{\#\text{scored claims}}, \quad (2)$$

$$\text{Relevance} = \frac{\#\text{claims labeled relevant}}{\#\text{scored claims}}. \quad (3)$$

Both scores are rounded to four decimals; if no claim can be scored, the corresponding value is `None`.

D Human Validation

Human annotators were given sampled commentary units or intermediate evaluation items with the corresponding game context when applicable. For type-aware decomposition, they independently assigned each unit one of three labels: descriptive, analytical, or background. For component validation, they independently annotated the target event, matching relation, reasoning relation, verification label, or background-claim judgment according to the provided context and annotation definitions. Their annotations were then compared with the LLM-produced outputs to compute agreement. Human validation was conducted by trained internal annotators following predefined annotation guidelines. No external crowdworkers or paid participant pools were recruited, no monetary compensation was provided. Annotators were instructed to rely only on the provided materials, and no personal information about annotators was collected.

Table 6: **Cross-judge consistency for analytical metrics.** We compare model rankings produced by the main GPT-5.4-mini-based judge and an alternative Gemini-3-Flash-based judge on a held-out subset.

Type	Metric	Spearman ρ
Analytical	Faithfulness	0.77
	Reasoning Depth	0.94

We conduct additional human validation for the LLM-based type-aware decomposition step. For each game, we randomly sample 100 commentary units after decomposition and manually annotate their commentary types as descriptive, analytical, or background. The LLM-produced labels are then compared with the human annotations to compute accuracy.

As shown in Table 5, the decomposition step achieves consistently high accuracy across all games, ranging from 93.0% on Go to 98.0% on football and basketball. The overall accuracy reaches 95.8%, indicating that the type-aware decomposition used by TACE is reliable across heterogeneous game domains.

E Cross-Judge Consistency Check

Because TACE uses GPT-5.4-mini for several LLM-mediated evaluation steps, we further examine whether the main analytical results are driven by judge-model preference. This concern is most relevant for analytical commentary: in descriptive evaluation, the key event-matching step is performed by GLM-4.7 rather than a GPT-based judge; in background evaluation, the best-performing models are not GPT-family models. In contrast, analytical node and edge verification are implemented with GPT-5.4-mini, while GPT-5.4 achieves the strongest analytical performance in the main results. We therefore conduct a targeted cross-judge consistency check for analytical metrics.

Specifically, on a held-out subset, we replace the GPT-5.4-mini-based node and edge verification judge with Gemini-3-Flash, while keeping the rest of the TACE pipeline unchanged. We then compare the resulting model rankings with those produced by the main judge using Spearman rank correlation. As shown in Table 6, the rankings remain consistent, especially for Reasoning Depth. This suggests that the analytical conclusions are not solely driven by preference from the GPT-based judge.

F Detailed Prompts

F.1 Prompts for Data Processing

Prompt A.1: Football sentence segmentation and type classification

You are a professional football commentary analysis expert. Your task is to precisely segment football commentary text into individual sentences and classify each sentence semantically.

Football Commentary Classification Standards.

1. **Descriptive Commentary (ID: 1)**
   - **Core Question:** "What is happening on the pitch?"
   - **Characteristics:** Objective narration of events and actions directly observable in the game.
   - **Includes:** Player movements (shooting, passing, dribbling, defending), score changes, referee calls, possession changes, etc.
   - **Examples:** ... ...
2. **Analytical Commentary (ID: 2)**
   - **Core Question:** "Why is this happening? What will happen next?"
   - **Characteristics:** Explains tactical intentions, evaluates decision quality, predicts game developments, analyzes technical details.
   - **Includes:** Tactical analysis, decision evaluation, situation prediction, technical movement breakdown.
   - **Examples:** ... ...
3. **Background Commentary (ID: 3)**
   - **Core Question:** "What is the background and significance of this event?"
   - **Characteristics:** Introduces external knowledge such as player backgrounds, historical data, version information, off-pitch stories, etc.
   - **Includes:** Player career history, head-to-head records, injury status, transfer background, statistical data.
   - **Examples:** ... ...

Important Rules.

- Must accurately split the original text into complete sentences; do not merge or over-split.
- Each sentence can only be assigned one most appropriate category.
- Output must be in strict JSON array format, with each element as a binary tuple `[sentence, category_id]`.
- Only output the JSON array, with no other text or explanations.

User Prompt Template.

```
Please analyze the following football commentary text:

{commentary}

Perform sentence segmentation and classification according to the above standards, and output the
JSON array.
```

Prompt A.2: Basketball sentence segmentation and type classification

You are a professional basketball commentary analysis expert. Your task is to precisely segment basketball commentary text into individual sentences and classify each sentence semantically.

Basketball Commentary Classification Standards.

1. **Descriptive Commentary (ID: 1)**
   - **Core Question:** "What is happening on the court?"
   - **Characteristics:** Objective narration of events and actions directly observable in the game.
   - **Includes:** Player movements (shooting, driving, passing, defending), score changes, referee calls, possession changes, etc.
   - **Examples:** ... ...
2. **Analytical Commentary (ID: 2)**
   - **Core Question:** "Why is this happening? What will happen next?"
   - **Characteristics:** Explains tactical intentions, evaluates decision quality, predicts game developments, analyzes technical details.
   - **Includes:** Tactical analysis, decision evaluation, situation prediction, technical movement breakdown.
   - **Examples:** ... ...
3. **Background Commentary (ID: 3)**

- **Core Question:** "What is the background and significance of this event?"
- **Characteristics:** Introduces external knowledge such as player backgrounds, historical data, version information, off-court stories, etc.
- **Includes:** Player career history, head-to-head records, injury status, trade background, statistical data.
- **Examples:** ... ...

Important Rules.

- Must accurately split the original text into complete sentences; do not merge or over-split.
- Each sentence can only be assigned one most appropriate category.
- Output must be in strict JSON array format, with each element as a binary tuple `[sentence, category_id]`.
- Only output the JSON array, with no other text or explanations.

User Prompt Template.

```
Please analyze the following basketball commentary text:

{commentary}

Perform sentence segmentation and classification according to the above standards, and output the
JSON array.
```

Prompt A.3: Caption-grounded sentence filtering (football)

Analyze each sentence independently, applying minimal paraphrasing only to resolve ambiguous references or spelling errors.

Retention Criteria (ALL must be satisfied):

1. **Reference Clarity:** Pronouns must clearly refer to a specific entity inferable from caption and **previously retained** commentary. **Pronoun inference must be completely unambiguous and free of any doubt.**
   - **Mandatory first-sentence truncation check:** Reject if the opening sentence appears incomplete (e.g., begins with conjunction/preposition/subordinate clause or lacks subject/verb). **In the first sentence, any pronoun referring to a player or team with uncertainty must be rejected, as it likely refers to preceding context that was truncated and cannot be definitively confirmed.**
2. **Relevance:** Content must directly relate to caption's action, event, or mentioned players.
   - **Category 1 (Action):** Describe caption's action or directly related on-field situations.
   - **Category 2 (Strategy):** Analyze tactics based on caption's action.
   - **Category 3 (Background):** Provide supplemental info ONLY for players appearing in caption; **MUST clearly and unambiguously refer to a specific player or team through inference that is completely unambiguous and free of doubt.**
3. **Semantic Consistency:** Must not contradict caption or retained sentences.
4. **Minimal Correction:** Rephrase solely for:
   - Ambiguous but inferable references.
   - Obvious name spelling errors (common in audio transcription).
   - Preserve original tone and intent.

Core Principles:

- Caption is authoritative; do not question its accuracy.
- **Only retained sentences may serve as context for subsequent evaluations**.
- **Background information must have a clearly inferable target; reject if the player/team reference is ambiguous or uncertain**.
- Violation of criteria 1–3 → reject without paraphrasing.

Output Format.

Part 1: Analysis. For each sentence, detail:

- Reference clarity (with truncation assessment and **pronoun ambiguity evaluation, especially for player/team references in first sentence**).
- Relevance and factual accuracy (state reason if failed; for Category 3, **explicitly flag target specificity and inference clarity concerns**).
- Semantic consistency (identify contradictions if failed).
- **Reference Specificity (Category 3 ONLY):** Confirm background information clearly and unambiguously refers to a specific player or team through inference that is completely unambiguous and free of doubt; reject if the target is unclear or requires ambiguous inference.

- Paraphrasing need (specify changes and rationale).
- Final verdict (retain/reject; provide corrected version if applicable).

Part 2: JSON.

```
[
  ["Sentence1", Category1, true/false],
  ["Sentence2", Category2, true/false],
  ...
]
```

- Strict array of triplets: `[sentence, category, retention flag]`.
- Sentence: original if unchanged, corrected if paraphrased.
- Retention flag: lowercase `true/false`.
- Category number unchanged.

User Prompt Template.

```
Caption (authoritative action description):
{caption}

Commentary Sentences (pre-classified):
{sentences_info}

Apply the specified retention criteria to evaluate each sentence independently. Output Part 1:
Analysis and Part 2: JSON.
```

Prompt A.4: Football caption-commentary fusion

You are an expert soccer video content analyst. Your task is to combine the caption and commentary into a coherent narrative written in the voice of a professional sports commentator.

Input:

- Caption: A brief textual description of a soccer/football game clip, focusing on the main events and player actions occurring in the video.
- Commentary: Transcribed live commentary from the same video segment, which may contain multiple pieces of information.

Task: Combine the caption and commentary into a coherent, natural-sounding description that captures the key elements of both, written **in the live broadcast voice while maintaining stylistic fidelity to the provided commentary—mirroring its energy, pacing, vocabulary, and tonal qualities**. The result should read like a unified live commentary of the event, not as two separate pieces of information.

Critical Filtering Rule: Before combining, carefully evaluate each piece of information in the commentary according to these strict hierarchical requirements:

1. **Event Description Criterion:** Commentary elements describing on-field actions must demonstrate direct relevance to the caption's events OR establish a clear chronological relationship with them. Exclude isolated event descriptions that lack temporal or causal connection to the caption narrative.
2. **Tactical Analysis Criterion:** Strategic or tactical observations must be explicitly anchored to specific event descriptions from the caption or retained commentary. Exclude standalone tactical analysis that references no concrete on-field action present in the source materials.
3. **Background Information Criterion:** Contextual details (player positioning, formations, situational factors) may only be included when they directly supplement and illuminate events described in the caption or retained commentary. Exclude background information wholly detached from the depicted action.
4. **General Enhancement Criterion:** Only retain commentary elements that satisfy the above three criteria and demonstrably enhance or complement the caption's narrative while maintaining absolute factual consistency.

Strict Factual Boundaries: Your entire narrative must be constructed **exclusively** from information explicitly present in the caption and commentary. You are prohibited from:

- Introducing any external facts, statistics, or context not provided in the source materials.
- Speculating on player motivations, emotions, or unspoken thoughts.
- Fabricating details about game situation (score, time, half) beyond stated information.
- Assuming or inventing player names, teams, venues, or historical data.
- Creating hypothetical scenarios or "what-if" scenarios.

Focus on:

1. Creating a seamless narrative **that authentically reflects the source commentary's unique style, delivery patterns, and rhetorical approach**, incorporating both factual events (caption) and relevant descriptive/color commentary.

2. Maintaining strict chronological flow of events as presented in the caption.
3. **Employing the specific language patterns, excitement cadence, and terminology preferences exhibited in the provided commentary**, ensuring stylistic consistency while applying professional broadcasting standards.
4. Ensuring clarity, readability, and factual accuracy by filtering out irrelevant or contradictory commentary.
5. Preserving all important information from the caption while selectively incorporating valuable commentary elements that enhance the broadcast-style narrative.
6. Maintaining absolute fidelity to the source material, with zero speculation or fabrication.

Output Format: Present your combined narrative inside a box using the format `\boxed{your_combined_narrative_here}`. This should be the only content in your response. The narrative must be written **in the distinctive style and tone captured in the source commentary**.

User Prompt Template.

```
Caption: {caption}
Commentary:
{commentary_text}
```

Prompt A.5: Basketball caption-commentary relevance gate

You are an expert basketball video content analyst. Your task is to determine whether a live commentary segment is semantically relevant to the current video content described in a caption.

Input:

- Caption: A brief textual description of a basketball game clip, focusing on the main events and player actions occurring in the video (e.g., "Player X shoots a three-pointer", "Team Y executes a fast break").
- Commentary: Transcribed live commentary from the same video segment, which may have temporal lag and could discuss events beyond the immediate clip.

Relevance Criteria: The commentary is considered RELEVANT (true) if:

1. It directly describes or comments on the events/actions mentioned in the caption, OR
2. It provides strategic analysis of team tactics based on the caption's events, OR
3. It offers background context about players/teams mentioned in the caption (e.g., season statistics, career achievements, team history), OR
4. It discusses game implications (score changes, momentum shifts) related to the caption's events.

The commentary is considered IRRELEVANT (false) if:

1. It discusses completely unrelated events (different plays, previous games, off-court topics), OR
2. It contains generic filler content with no connection to the caption's specific events or entities.

Output Format.

- **Phase 1 (Analysis):** Provide a concise reasoning analyzing the semantic relationship between caption and commentary. Consider temporal context, entity mentions, and thematic connections.
- **Phase 2 (Judgment):** Output ONLY the final judgment as `\boxed{true}` or `\boxed{false}` with no additional text.

Important: Your analysis must be objective and based solely on semantic relevance. Do not over-interpret vague connections. The boxed judgment must be your definitive conclusion.

User Prompt Template.

```
caption: {caption}
commentary: {commentary}
```

Prompt A.6: Basketball caption-commentary fusion

You are an expert basketball video content analyst. Your task is to combine the caption and commentary into a coherent narrative written in the voice of a professional sports commentator.

Input:

- Caption: A brief textual description of a basketball game clip, focusing on the main events and player actions occurring in the video.
- Commentary: Transcribed live commentary from the same video segment, which may contain multiple pieces of information.

Task: Combine the caption and commentary into a coherent, natural-sounding description that captures the key elements of both, written **in the live broadcast voice while maintaining stylistic fidelity to the provided commentary—mirroring its energy, pacing, vocabulary, and tonal qualities**. The result should read like a unified live commentary of the event, not as two separate pieces of information.

Critical Filtering Rule: Before combining, carefully evaluate each piece of information in the commentary:

- If a commentary element is directly supported by or consistent with the caption's events, include it.

- If a commentary element is unrelated to the caption's described events, exclude it.
- If a commentary element contradicts the factual events described in the caption, exclude it.
- Only retain commentary elements that enhance or complement the caption's narrative.

Strict Factual Boundaries: Your entire narrative must be constructed **exclusively** from information explicitly present in the caption and commentary. You are prohibited from:

- Introducing any external facts, statistics, or context not provided in the source materials.
- Speculating on player motivations, emotions, or unspoken thoughts.
- Fabricating details about game situation (score, time, quarter) beyond stated information.
- Assuming or inventing player names, teams, venues, or historical data.
- Creating hypothetical scenarios or "what-if" scenarios.

Focus on:

1. Creating a seamless narrative **that authentically reflects the source commentary's unique style, delivery patterns, and rhetorical approach**, incorporating both factual events (caption) and relevant descriptive/color commentary.
2. Maintaining strict chronological flow of events as presented in the caption.
3. **Employing the specific language patterns, excitement cadence, and terminology preferences exhibited in the provided commentary**, ensuring stylistic consistency while applying professional broadcasting standards.
4. Ensuring clarity, readability, and factual accuracy by filtering out irrelevant or contradictory commentary.
5. Preserving all important information from the caption while selectively incorporating valuable commentary elements that enhance the broadcast-style narrative.
6. Maintaining absolute fidelity to the source material, with zero speculation or fabrication.

Output Format: Present your combined narrative inside a box using the format `\boxed{your_combined_narrative_here}`. This should be the only content in your response. The narrative must be written **in the distinctive style and tone captured in the source commentary**.

User Prompt Template.

```
Caption: {caption}
Commentary: {commentary}
```

Prompt A.7: Basketball quality selection prompt

System Prompt.

You are a professional sports-commentary analyst. From a batch of basketball game commentaries, select the two highest-quality items.

Evaluation Criteria: Each commentary will be scored on a 10-point scale across the following three dimensions, and the final ranking is based on the total score:

1. **Descriptive Dimension (Descriptive):** Evaluate the precision and information density with which the commentary captures the real-time game picture: whether observable facts such as the ball-handler, tactical movement, screen quality, shot area, defensive matchup, and referee calls are stated clearly, completely, and in detail. A higher score means the account of "what is happening at this moment" is more like a precise courtside documentary.
2. **Analytical Dimension (Analytical):** Evaluate the depth of interpretation of the game's underlying logic: whether offensive actions (Spain P&R, Horns, Flare), defensive strategies (switching, double-teaming, ICE), player decision quality, coaching matchups, and tactical trade-offs under score/time/foul situations are analyzed thoroughly. A higher score means the insight into "why the play unfolds this way" is more like that of a tactical master.
3. **Background Dimension (Background):** Evaluate the ability to extend the event context: whether player historical statistics, matchup histories, team season trends, key-game conventions, cultural references (e.g., "Christmas Day game"), sneakers, records, and other off-court information are naturally integrated to add the "deeper meaning of this moment" to the current picture. A higher score means the rendering of "what this moment means" is more like that of a storyteller.

Analysis Requirements: Follow these steps:

- **Step 1 (Analysis Phase):** For each commentary, provide analysis and a score for each of the three dimensions on a 10-point scale, then calculate the total score.
- **Step 2 (Selection Phase):** Sort the commentaries by total score from high to low, select the two highest-scoring commentaries, and return only their `video_id` values.

Output Format:

```
[Analysis Phase]
1. [video_id]:
   - Descriptive Dimension (Descriptive): [Reason] - X points
   - Analytical Dimension (Analytical): [Reason] - X points
   - Background Dimension (Background): [Reason] - X points
```

```
   - Total Score: XX points

...

[Selection Phase]
["video_id1", "video_id2"]
```

User Prompt Template.

```
Select the two highest-quality commentaries from the following 3 basketball-game commentaries:

{commentary_list}
```

F.2 Prompts for Commentary Generation

Prompt B.1: Football / Basketball candidate-generation prompt

Role: You are a top-tier professional football / basketball commentator. Your task is to generate an immersive, high-quality live commentary track based on the provided description of a match video clip.
Core Objective: Produce a coherent, pure-text narrative that replicates the energy, pacing, and terminology of a live TV broadcast. The output must be spoken directly to the audience (first-person active voice), avoiding third-party summarization styles.
Content Requirements (Semantic Layers): Your commentary must align with the three core commentator capabilities and their corresponding commentary types:

1. **Live Observation → Descriptive Commentary:**
   - Accurately perceive and interpret real-time events and fine-grained details in the game environment.
   - Objectively answer "What is happening?" by narrating immediate, observable events.
2. **Strategic Analysis → Analytical Commentary:**
   - Comprehend the current game state, interpret strategic intent, evaluate decisions, and anticipate future developments.
   - Go beyond description by assessing "How good?", explaining "Why?", and predicting "What's next?" within a broader tactical context.
3. **Historical Recall → Background Commentary:**
   - Leverage extensive game knowledge, including player histories, classic match moments, and community-specific terminology.
   - Answer "What is the significance?" by integrating broader contextual information such as detailed team and player backgrounds, game history, and cultural references.

Style & Tone Guidelines:

- **Dynamic Pacing:** Match the rhythm of the text to the intensity of the event (e.g., short, punchy sentences for goals; longer, analytical flows for build-up play).
- **Professional Vocabulary:** Use authentic football / basketball terminology.
- **Natural Fluency:** Avoid robotic sentence structures. Use varied syntax and emotive language characteristic of human commentators.

Strict Constraints:

1. **Format:** Output raw text only as a **single continuous paragraph**. **Do not use line breaks** (\n), Markdown, bolding, labels, or list formats.
2. **Language:** Maintain a single language consistently. Use **English** uniformly.
3. **Factual Grounding:** Do not hallucinate scorelines, team names, or events not implied by the input. Do not invent specific statistics unless provided.
4. **Consistency:** Ensure the narrative flow is logical and non-contradictory.
5. **Scope:** The video is only a segment of the match. The commentary must focus exclusively on this segment, not the full game.

User Prompt Template.

```
Match Background (Full Match Context, not limited to this clip):
{background}

Generate a football / basketball commentary based on the video content. The video segment is
{duration_seconds} seconds long. Complete the entire commentary within {duration_seconds} seconds.
```

Prompt B.2: Chess / Go candidate-generation prompt

System Prompt. You are a top-tier professional chess / go commentator. Your task is to generate an immersive, high-quality move commentary based on the provided game state information.
User Prompt Template.
Role: You are a top-tier professional chess / go commentator. Your task is to generate an immersive, high-quality move-by-move commentary based on the provided game state.
Core Objective: Produce a coherent, pure-text narrative that replicates the insight, pacing, and terminology of a professional chess / go broadcast. The output must be spoken directly to the audience (first-person active voice), avoiding third-party summarization styles.
Content Guidelines (Semantic Layers): You may draw on the following three core commentary capabilities of good commentators. These are provided as flexible guidance rather than requirements, and may be blended naturally.

1. **Move Observation → Descriptive Commentary:**
   - Accurately perceive and describe the current move, the piece involved, and the squares it moves from and to.
   - Objectively answer "What is happening on the board?" by narrating the immediate, observable move.
2. **Strategic Analysis → Analytical Commentary:**
   - Comprehend the current board position (provided via FEN), interpret the tactical or positional intent behind the move, evaluate its quality, and anticipate future developments.
   - Go beyond description by assessing "How good is this move?", explaining "Why was it played?", and predicting "What's next?" within a broader strategic context.
3. **Historical Recall → Background Commentary:**
   - Leverage extensive chess / go knowledge, including opening theory, classic game references, and player tendencies.
   - Answer "What is the significance?" by integrating broader contextual information such as player backgrounds, tournament stakes, and historical parallels.

Style & Tone Guidelines:

- **Professional Vocabulary:** Use authentic chess / go terminology.
- **Natural Fluency:** Avoid robotic sentence structures. Use varied syntax and emotive language characteristic of human commentators.

Strict Constraints:

1. **Format:** Output raw text only as a **single continuous paragraph**. **Do not use line breaks** (\n), Markdown, bolding, labels, or list formats.
2. **Language:** Use **English** uniformly.
3. **Factual Grounding:** Do not hallucinate player names, ratings, or events not implied by the input. Do not invent specific statistics unless provided.
4. **Consistency:** Ensure the narrative flow is logical and non-contradictory.
5. **Scope:** The commentary can focus on this specific move and its immediate context or the entire game.

```
Match Background (Full Game Context):
{background}

Game State:
Board before current move (FEN): {fen_before}
Current move: {full_move_str} (SAN={san}, UCI={uci})
Board after current move (FEN): {fen_after}
Previous moves (up to last 3 full moves by both sides): {history_text}


Game State:
Board before current move (X=Black, O=White, .=empty):
{board_ascii}
Current move (ply {ply}): {move_str} (SGF coord={coord_sgf}, standard={coord_std})
Recent moves (last 3 full rounds): {history_text}

Generate a chess / go commentary for this move.
```

Prompt B.3: League of Legends candidate-generation prompt

System Prompt. You are a top-tier professional League of Legends commentator. Your task is to generate an immersive, high-quality live commentary track based on the provided match video clip.
User Prompt Template.
Role: You are a top-tier professional League of Legends commentator. Your task is to generate an immersive, high-quality live commentary track based on the provided match video clip.
Core Objective: Produce a coherent, pure-text narrative that replicates the energy, pacing, and terminology of a live esports

broadcast. The output must be spoken directly to the audience (first-person active voice), avoiding third-party summarization styles.
Content Guidelines (Semantic Layers): You may draw on the following three core commentary capabilities of good commentators. These are provided as flexible guidance rather than requirements, and may be blended naturally.

1. **Live Observation → Descriptive Commentary:**
   - Accurately perceive and interpret real-time events and fine-grained details in the game environment.
   - Objectively answer "What is happening?" by narrating immediate, observable events (champion movements, ability usage, objective takes, kills, map states).
2. **Strategic Analysis → Analytical Commentary:**
   - Comprehend the current game state, interpret strategic intent, evaluate decisions, and anticipate future developments.
   - Go beyond description by assessing "How good?", explaining "Why?", and predicting "What's next?" within a broader tactical context (draft implications, win conditions, power spikes, macro decisions).
3. **Historical Recall → Background Commentary:**
   - Leverage extensive League of Legends knowledge, including player histories, classic match moments, and community-specific terminology.
   - Answer "What is the significance?" by integrating broader contextual information such as team and player backgrounds, game history, and cultural references.

Style & Tone Guidelines:

- **Professional Vocabulary:** Use authentic League of Legends terminology (e.g., "lane pressure," "reset," "trade," "skirmish," "rotation," "vision control," "engage," "disengage," "objective posture," "power spike," "team comp").
- **Natural Fluency:** Avoid robotic sentence structures. Use varied syntax and emotive language characteristic of human casters.

Strict Constraints:

1. **Format:** Output raw text only as a **single continuous paragraph**. **Do not use line breaks** (\n), Markdown, bolding, labels, or list formats.
2. **Language:** Use **English** uniformly.
3. **Factual Grounding:** Do not hallucinate champion names, gold values, objectives, kills, or map states that are not clearly supported by the video or provided context.
4. **Consistency:** Ensure the narrative flow is logical and non-contradictory.
5. **Scope:** The video is only a segment of the match. The commentary must focus exclusively on this segment, not the full game.

```
Match Background (Full Match Context, not limited to this clip):
{background}

{extra_context_block}

The video segment is {duration_seconds} seconds long. Complete the entire commentary within
{duration_seconds} seconds.

Generate a League of Legends commentary based on the video content.
```

F.3 Prompt for TACE

F.3.1 Prompt for Generated Commentary Segmentation

Prompt C.1: Evaluation segmentation and type classification (football)

System Prompt. You are a professional football commentary analysis expert. Your task is to precisely segment football commentary text into individual sentences and classify each sentence semantically.
Football Commentary Classification Standards.

1. **Descriptive Commentary (ID: 1)**
   - **Core Question:** "What is happening on the pitch?"
   - **Characteristics:** Objective narration of events and actions directly observable in the game, without subjective evaluation.
   - **Includes:** Player movements (shooting, passing, dribbling, defending), score changes, referee calls, possession changes, etc.
   - **Examples:** ... ...
2. **Analytical Commentary (ID: 2)**
   - **Core Question:** "How good? Why is this happening? What will happen next?"

- **Characteristics:** Subjectively evaluates performance or decisions, explains tactical intentions and causes, predicts game developments, analyzes technical details.
- **Includes:** Tactical analysis, decision evaluation, situation prediction, technical movement breakdown, emotional evaluation of plays.
- **Examples:** ... ...

3. **Background Commentary (ID: 3)**
   - **Core Question:** "What is the background and significance of this event?"
   - **Characteristics:** Introduces external knowledge such as player backgrounds, historical data, off-pitch stories, etc.
   - **Includes:** Player career history, head-to-head records, injury status, transfer background, statistical data.
   - **Examples:** ... ...

Boundary Rules:
- When a sentence contains elements of multiple categories, classify based on its primary intent (semantic focus).
- Pure emotional exclamations with evaluative tone (e.g., "What a goal!") belong to Analytical Commentary (ID: 2).
- Objective data statements (e.g., "This is his 30th goal") belong to Background Commentary (ID: 3); interpretation or inference drawn from data belongs to Analytical Commentary (ID: 2).

Important Rules:
- Segment text into the smallest independent semantic units (clauses or short phrases) rather than full grammatical sentences. Split compound sentences at commas or conjunctions if they contain distinct events or judgments.
- Each sentence can only be assigned one most appropriate category.
- Output must be in strict JSON array format, with each element as a binary tuple `[sentence, category_id]`.
- Only output the JSON array, with no other text or explanations.

User Prompt Template.

```
Please analyze the following football commentary text:

{commentary}

Perform sentence segmentation and classification according to the above standards, and output the
JSON array.
```

F.3.2 Prompt for Descriptive Evaluation

Prompt C.2: Reference event extraction (football)

System Prompt. You are a football event extraction expert. Your task is to extract the main events from the given commentary and label information.

Input:
- `label`: The main event type of the video clip (e.g., goal, corner, clearance, etc.).
- `processed_commentary`: The processed reference commentary describing what happened.
- `background_xml`: Full XML-formatted match background, including teams and player rosters.

Task: Extract the main events that occurred in the video clip. Each event should be described in a single sentence including:
- Who (player name or team).
- What (action).
- Result (if applicable).

Output Format.

```
["Event 1 description", "Event 2 description", ...]
```

Important:
- Only output the JSON array.
- All event descriptions must be in English.
- Focus on the main events mentioned in the label and commentary.
- Do not include minor details or repetitive descriptions.

Examples. ... ...

User Prompt Template.

```
Label: {label}
Processed Commentary: {processed_commentary}
```

```
Background XML:
{background}

Extract the main events as a JSON array of strings.
```

Prompt C.3: Candidate event extraction (football)

System Prompt. You are a football event extraction expert. Your task is to extract the main events from the given commentary text.
Task: Extract the main events that occurred in the video clip based on the commentary. Each event should be described in a single sentence including:

- Who (player name or team).
- What (action).
- Result (if applicable).

Output Format.

```
["Event 1 description", "Event 2 description", ...]
```

Important:

- Only output the JSON array.
- All event descriptions must be in English.
- Focus on the main events mentioned in the commentary.
- Do not include minor details or repetitive descriptions.

Examples. ... ...
User Prompt Template.

```
Commentary: {commentary}

Extract the main events as a JSON array of strings.
```

Prompt C.4: Completeness matching

System Prompt. You are a semantic matching expert for completeness evaluation. Your task is to match ground truth events to evaluated events.
Input:

- Ground Truth Events: The reference events that should be covered (from human-annotated data, high certainty).
- Evaluated Events: The events extracted from the commentary being evaluated.

Task: For each event in Ground Truth Events, determine if there is a matching event in Evaluated Events.
Strict Matching Criteria:

- The action must be highly consistent.
- The subject (player/team) must be clearly identified and match.
- The result must match (if applicable).
- Only match if you are confident they describe the same event.

Output Format.

```
{
  "matches": [evaluated_index_or_null, ...],
  "reasoning": "detailed explanation of matching decisions"
}
```

matches array:

- Length must equal Ground Truth Events length.
- Each element is the index (0-based) of the matching event in Evaluated Events, or `null` if no match.

Examples. ... ...
User Prompt Template.

```
Ground Truth Events: {ground_truth_events}
Evaluated Events: {evaluated_events}

For each ground truth event, find the matching evaluated event index (or null). Return as JSON
with "matches" array and "reasoning".
```

Prompt C.5: Correctness matching

System Prompt. You are a semantic matching expert for precision evaluation. Your task is to match evaluated events to ground truth events.
Input:

- Evaluated Events: The events extracted from the commentary being evaluated.
- Ground Truth Events: The reference-text-derived event list, which is considered accurate and should be treated as the evidence source.

Task: For each event in Evaluated Events, determine if there is a matching event in Ground Truth Events.
Moderately Strict, Evidence-Conservative Matching Criteria:

- The core action must match. Do not match different event categories just because they happen in a similar scene.
- The result or outcome must also match when applicable (for example: score vs miss, save vs goal, foul vs clean tackle).
- The subject, team, and attacking side must not conflict.
- Flexible references are allowed when they can plausibly refer to the same actor without conflict, such as names, jersey numbers, roles, or descriptive references.
- Generalization is allowed if the evaluated event is less specific but still supported by the ground truth event.
- Do not match if the evaluated event adds unsupported critical details that are not grounded in the reference event, such as a specific player, team, attacking side, event type, decisive result, or key execution detail.
- Be conservative: if a critical fact is unsupported or conflicts with the evidence, return `null`.

Output Format.

```
{
  "matches": [gt_index_or_null, ...],
  "reasoning": "detailed explanation of matching decisions"
}
```

matches array:

- Length must equal Evaluated Events length.
- Each element is the index (0-based) of the matching event in Ground Truth Events, or `null` if no match.

Examples. ... ...
User Prompt Template.

```
Evaluated Events: {evaluated_events}
Ground Truth Events: {ground_truth_events}

For each evaluated event, find the matching ground truth event index (or null). Return as JSON
with "matches" array and "reasoning".
```

F.3.3 Prompt for Analytical Evaluation

Prompt C.6: Analytical relation standard

- **CAUSE:** A directly brings about, triggers, or produces B.
- **REASON:** B provides the cause, explanation, evidence, or justification for A.
- **CONDITION:** A states a hypothetical or unrealized premise; B holds only under that premise.
- **CONTRAST:** A establishes an expectation or direction; B violates or reverses it.
- **ELABORATION:** B is an intrinsic structural component or unpacking of A.

Prompt C.7: Analytical proposition extraction

System Prompt.

User Prompt. You extract analytical propositions from stage-1 analytical sentences.
Task:

- Use the full commentary only as context for reference resolution.
- Extract propositions ONLY from the provided analytical sentence records.
- Each proposition must remain inside its own source sentence.
- Never fuse content across two different analytical sentences.

- Preserve sentence order.
- Preserve proposition order inside each sentence.
- Prefer complete proposition units, not fragments.

Input Template.

```
Input commentary:
{commentary}

Analytical sentence records:
{analytical_sentence_records_json}
```

Output requirements:

- Output exactly one strict JSON array and nothing else.
- The array length must equal the number of analytical sentence records.
- Each item must be an object with:
  - `source_text`
  - `propositions`
- `propositions` must be a non-empty JSON array of strings.

Example output.

```
[
  {
    "source_text": "This is a very practical decision",
    "propositions": ["This is a very practical decision"]
  }
]
```

Prompt C.8: Analytical edge extraction

System Prompt.

User Prompt. You analyze an analytical proposition batch and output only the sparse directed edges whose true anchor is the current source proposition.

Relation type definitions.

```
{relation_standard}
```

Rules:

- Use the full batch for context, but only output edges whose `source_proposition_id` equals the requested source proposition.
- Consider only forward edges to later propositions in the provided batch order.
- Treat all propositions uniformly; do not use sentence-boundary reasoning.
- Keep only clear, necessary, information-bearing edges.
- Do not output redundant shortcuts when a better downstream anchor exists.
- Do not output edges based only on same topic/entity mention or loose continuity.
- Output at most one relation per source-target pair.
- If no edge should be kept, output `[]`.

Input Template.

```
Current source proposition id:
{source_proposition_id}

Proposition batch:
{proposition_batch_json}
```

Output format.

```
[
  {
    "source_proposition_id": "ap_0000",
    "target_proposition_id": "ap_0003",
    "relation": "CAUSE",
    "explanation": "One short plain-text sentence only."
  }
]
```

Prompt C.9: Analytical edge audit

System Prompt.

User Prompt. You audit one candidate edge in an analytical proposition DAG.
Relation type definitions.

{relation_standard}

Keep the candidate edge only if BOTH are clearly true:

1. Correctness: the direction and relation are directly supported by the propositions.
2. Necessity: the edge is necessary in the final sparse DAG, not a redundant shortcut or weak thematic link.

Guidance:

- Treat all proposition pairs uniformly; do not rely on sentence-boundary reasoning.
- Delete edges that are only temporal adjacency, same-topic mention, or narrative continuation.
- Delete broader shortcuts when a better downstream anchor already captures the relation.
- If not clearly convinced, delete.

Allowed reason codes: ...
Input Template.

```
Full proposition list:
{all_propositions_json}

Draft edge list:
{draft_edges_json}

Candidate edge:
{candidate_edge_json}
```

Output exactly one JSON object and nothing else.

```
{
  "keep": true,
  "is_correct": true,
  "is_necessary": true,
  "reason_code": "KEEP",
  "explanation": "One short plain-text sentence only."
}
```

Prompt C.10: Analytical node and edge truth judgment

System Prompt.

User Prompt. You judge whether each analytical proposition and each final DAG edge is true/supported in the commentary context.
Important:

- Judge node/proposition truth and edge correctness, not extraction faithfulness.
- For an edge, use the given source node, target node, and relation together.
- Use the reference-event list as the primary evidence.
- Use the commentary only as necessary context for reference resolution.

Mark a node true when:

- the proposition is supported by the reference-event list;
- the commentary context helps resolve pronouns or omitted referents without changing the claim;
- the proposition does not add unsupported event facts beyond the reference-event list.

Mark a node false when:

- the proposition contradicts the reference-event list;
- the proposition depends on event facts that are not supported by the reference-event list;
- the commentary context does not supply enough support to make the proposition true.

Mark an edge true only when:

- both endpoint propositions are supported;
- the directed relation and relation type are supported by the two propositions plus reference-event context.

Mark an edge false when:

- either endpoint proposition is unsupported;
- the direction or relation type is unsupported;
- the link is only loose thematic, temporal, or narrative continuity.

Input Template.

```
Commentary:
{commentary}

Candidate propositions:
{candidate_propositions_json}

Candidate final DAG edges:
{candidate_edges_json}

Reference-event list:
{reference_events_json}
```

Output requirements:

- Output exactly one strict JSON object and nothing else.
- `node_labels` length must equal the number of candidate propositions.
- `edge_labels` length must equal the number of candidate final DAG edges.

Example output.

```
{"node_labels": [true, false, true], "edge_labels": [false, true]}
```

F.3.4 Prompt for Background Evaluation

Prompt C.11: Background claim extraction (football)

System Prompt. You are a football background-claim extraction expert.
Extract medium-granularity background claims from football background commentary.

Rules:

- Extract only claims supported by the source sentence.
- Use context only for disambiguation, entity resolution, and time-scope completion.
- Do not invent unsupported claims.
- Return self-contained, independently judgeable background claims.
- If no valid background claim exists, return no claim for that sentence.

Allowed `claim_type` values:

- `identity_profile`
- `affiliation_history`
- `statistical_claim`
- `form_assessment`
- `match_metadata`
- `event_significance`
- `other_background`

Allowed `focus_key` values include: ... ...
Allowed `time_scope` values: ... ...
Each entity must contain `mention`, `resolved_name`, `entity_type`, and `resolution_confidence`. `resolution_confidence` should be `high` / `medium` / `low`.
Each claim object must include:

- `claim_id`
- `source_sentence_id`
- `source_sentence`
- `claim_text`
- `claim_type`

- focus_key
- entities
- time_scope

Add claimed_value when the claim explicitly contains a numeric/statistical target. Add qualifiers when the claim requires extra qualifiers to be judgeable.
In batch mode, source_sentence_id must be the zero-based index in All Background Sentences. In per_sentence mode, source_sentence_id must equal the Target Background Sentence ID. source_sentence must exactly copy the corresponding sentence text from All Background Sentences. Do not output legacy keys such as claim/type instead of claim_text/claim_type. Use [] when no valid background claim exists.
In batch mode, source_sentence_id must be the zero-based index in All Background Sentences. In per_sentence mode, source_sentence_id must equal the Target Background Sentence ID. source_sentence must exactly copy the corresponding sentence text from All Background Sentences. Do not output legacy keys such as claim/type instead of claim_text/claim_type. Use [] when no valid background claim exists.
Return a strict JSON array only.
User Prompt Template.

```
Extraction Mode: {extraction_mode}

Sport Type: {sport_type}

Target Background Sentence (ID={source_sentence_id}):
{target_background_sentence}

Neighbor Background Sentences:
{neighbor_background_sentences}

All Background Sentences:
{all_background_sentences}

Descriptive Clauses:
{descriptive_clauses}

Analytical Clauses:
{analytical_clauses}

Parsed Match Context:
{parsed_match_context}

Extract zero or more background claims.
Every claim object must use the required schema from the system prompt.
In batch mode, use the zero-based index from All Background Sentences as source_sentence_id.
source_sentence must exactly copy that indexed sentence.
Return a strict JSON array only.
```

Prompt C.12: Background factuality judgment

System Prompt. You are a factuality judge for background commentary claims.
Your task is to judge a single background claim as true or false.
Core rules:

- The local background KB evidence package is the primary evidence source.
- You may use common knowledge only as a supplement, never as a replacement for stronger KB evidence.
- Every claim must receive a binary label: true or false.
- Prefer false when the available evidence does not justify a true judgement.
- Always return evidence_ids from the KB evidence items you actually used.

Return strict JSON only with:

```
{
  "label": "true" or "false",
  "reasoning": "...",
  "evidence_ids": [...],
  "kb_priority_respected": true or false
}
```

User Prompt Template.

```
Claim:
{claim}

Evidence Package:
{evidence_package}
```

```
Judge the claim as true or false.
The local KB evidence package must be treated as primary evidence.
Return strict JSON only.
```

Prompt C.13: Background relevance judgment

System Prompt. You are a relevance matcher for background commentary claims.
Your task is to judge whether a background claim is relevant to the current clip, where the current clip is represented by descriptive clauses and analytical clauses.
Rules:

- A claim is relevant if it meaningfully supports, explains, or contextualizes at least one descriptive or analytical clause.
- A claim is irrelevant if it does not meaningfully support either clause set.
- A single claim may match multiple descriptive clauses and multiple analytical clauses.
- Return exact clause ids you matched.
- If there is no meaningful match, return empty id lists and `label=irrelevant`.

Return strict JSON only with:

```
{
  "label": "relevant" or "irrelevant",
  "matched_descriptive_sentence_ids": [...],
  "matched_analytical_sentence_ids": [...],
  "reasoning": "..."
}
```

User Prompt Template.

```
Claim:
{claim}

Descriptive Clauses:
{descriptive_clauses}

Analytical Clauses:
{analytical_clauses}

Local Background Context:
{local_background_context}

Judge whether this background claim is relevant.
Return strict JSON only.
```