# SAMA-GEO: DO SEMANTIC-AWARE MULTI-AGENT SYSTEMS WORK FOR GAME ENGINE OPTIMIZATION?

**Anonymous authors**
Paper under double-blind review

## ABSTRACT

Game engine optimization across multi-game platforms presents a challenging decision-making problem due to the heterogeneous and complex characteristics of game workloads, with significant variation across and within genres. While genre provides a useful high-level characterization of games, it alone is insufficient for accurate policy prediction, as substantial variation in gameplay characteristics and engine requirements exists even among games within the same genre. We ask: do semantic-aware multi-agent systems work for this problem? To enable this analysis, we introduce SAMA-GEO, a benchmark dataset of 5,193 Roblox games spanning 17 genres and 59 genre-subgenre pairs, constructed using publicly available Roblox APIs. The dataset features a gold set built through manual play-testing and expert annotation, and a larger silver set with gold-referenced annotations, with each game annotated with 10 semantic labels capturing gameplay characteristics and 3 policy labels governing engine configuration. We formulate game engine optimization on Roblox as semantic-aware policy prediction, where game-level semantic understanding provides an explicit intermediate signal for predicting engine policies, bridging the gap that genre alone cannot close. To investigate this formulation, we propose a Semantic-Aware Multi-Agent framework that decomposes semantic and policy reasoning across specialized agents, representing the first application of multi-agent reasoning to game engine optimization. To measure performance, we develop an eight-metric evaluation suite spanning semantic and policy prediction, combining established, adapted, and novel metrics that jointly capture pointwise accuracy, structural preservation, and constraint adherence. Component-wise analysis shows consistent and complementary gains as semantic and policy reasoning capabilities are incorporated, supporting an affirmative answer to our central question.

## 1 INTRODUCTION

Game engines rely on numerous configurable parameters that determine how a game runs. Choices over these parameters directly affect game performance on the engine and, consequently, the player experience. We consider an engine configuration policy to be optimal when it provides the best possible player experience for a given game. In this work, we focus on three key policy attributes that directly influence this experience: target FPS, resolution, and render quality. The optimal policy depends on a range of factors, including gameplay characteristics, genre, visual requirements, and system requirements. Consequently, multi-game platforms such as Roblox cannot rely on a single configuration policy across games, as the requirements of individual games can differ substantially. For example, a first-person shooter may prioritize a consistently high frame rate over visual fidelity due to its fast-paced gameplay, whereas a large-scale open-world adventure game may place greater emphasis on visual quality and scene complexity. Moreover, substantial variation can exist even among games within the same genre. Genre alone is therefore insufficient to determine optimal policy for individual games, motivating a richer characterization of each game's specific requirements.

The engine policy of a game is strongly influenced by its semantic profile. Although genre provides a useful high-level characterization of games(Apperley, 2006), it does not fully capture the variation in gameplay and system requirements across games(Clarke et al., 2017), including those within the same genre. Characteristics of how a game is presented, played and experienced can provide important information about the requirements imposed on its engine and, consequently, its optimal



Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

policy. For instance, a game's motion dynamics and frequency of player interaction can influence its frame-rate requirements, while visual complexity and physics behavior can affect its resolution and rendering configuration. Other semantic characteristics, such as typical session length and dependence on social interaction, can also influence different aspects of the overall engine policy. These relationships suggest that game semantics can serve as a meaningful intermediate representation between a game and its engine policy, and raise the question of whether a semantic-aware multi-agent system can leverage this structure to predict engine policies across heterogeneous game workloads.

To investigate this, we ground our analysis in Roblox, the world's largest multi-game platform, leveraging its publicly available APIs to construct SAMA-GEO, a benchmark for multi-game engine optimization. While several gaming-related benchmarks exist(Ji et al., 2025; Ouyang et al., 2026; Zheng et al., 2025; Zhang et al., 2025), none capture the breadth and heterogeneity of multi-game engine optimization. SAMA-GEO comprises 5,193 games spanning 17 genres and 59 genre-subgenre pairs, with a gold set of manually play-tested and expert-annotated games and a larger silver set with gold-referenced annotations. Each game is annotated with 10 semantic labels capturing gameplay characteristics and 3 policy labels governing engine configuration, with semantic labels translated to policy labels via a GMM-based pipeline verified by expert annotators. Complementing the dataset, we develop an eight-metric evaluation suite combining two established, five adapted, and one novel metric that jointly capture pointwise accuracy, structural preservation, and constraint adherence. Existing metrics assess whether predictions satisfy target constraints, but a model could satisfy constraints while producing semantically incoherent policies, assigning identical engine configurations to games with vastly different semantic profiles. To this end, we propose the Semantic-Policy Consistency Score (SPCS), which is a novel metric that measures whether predicted policies vary consistently with predicted semantic characteristics. SPCS captures this failure mode directly, complementing constraint-based metrics with a pairwise measure of semantic-policy alignment. The dataset and the suite form SAMA-GEO as a full benchmark for this problem.

Equipped with this benchmark, we formulate multi-game engine optimization as semantic-aware policy prediction. Rather than predicting engine policies directly, we first identify a game's semantic characteristics and use them to inform policy prediction. Specialized agents have been used to decompose complex tasks and coordinate complementary information(Hong et al., 2024; Wu et al., 2023), and the heterogeneous nature of game semantics similarly motivates specialized reasoning roles. We thus propose a multi-agent framework that leverages diverse sources of game information including genre, textual description, images, and metadata to establish a game's semantic profile which is used to predict its engine policy. Our framework comprises four agents, a Game Analyst, a Resource-Aware Semantic Coordination Refiner, a Genre Analysis Agent, and an Image Analysis Agent, representing the first application of multi-agent reasoning to game engine optimization.

Our main contributions can be summarized as follows:

- We introduce a novel problem of semantic-aware policy prediction for engine optimization of multi-game platforms
- We introduce the SAMA-GEO benchmark which includes
  - a two-tier dataset which is a collection of 5,193 Roblox games comprising a manually play-tested expert-annotated gold set, and a silver set with gold-referenced annotations
  - an eight-metric evaluation suite covering semantic and policy prediction, combining existing, adapted, and novel metrics including SPCS, a newly proposed metric that captures semantic-policy consistency, addressing a previously undetected failure mode
- We propose the SAMA-GEO framework, the first multi-agent system for game engine optimization, decomposing semantic and policy reasoning across specialized agents

## 2 Related Work

Multi-game engine policy prediction remains largely unexplored. However, machine learning has been widely applied to video game analysis and engine optimization, while multi-agent frameworks have emerged as a popular paradigm for decomposing and solving complex tasks. We briefly review relevant work below.

**Machine Learning for Games.** Machine learning has been widely applied to video games for game understanding, gameplay modeling, and performance analysis(Justesen et al., 2019; Smerdov et al.,




Figure 1: **Samples from SAMA-GEO dataset.** Showcase of 3 Roblox games from diverse genres and popularity scales. Each game has a genre-subgenre pair(subgenre can be blank, third example), description, images, no. of visits, max players, voice chat support and availability of private servers.

2023). Game-level representations have been used for tasks such as genre and gameplay classification(Vargas-Iglesias, 2020; Jiang & Zheng, 2023), while recent LLM- and VLM-based approaches enable richer understanding from textual and visual game information(Hu et al., 2026; Xie et al., 2026). Learning-based methods have also explored game performance and engine optimization, including rendering and system configuration(Heo et al., 2026; Mohebali & Chiew, 2014). However, existing work typically focuses on individual games, specific objectives, or isolated aspects of game behavior. As discussed, while several gaming datasets and benchmarks exist, none target the cross-game engine policy prediction setting considered here. Roblox provides a natural setting for this analysis, as its engine hosts a large number and variety of games across diverse genres, making it uniquely suited for studying cross-game engine optimization at scale.

**Multi-agent Frameworks.** Multi-agent frameworks have recently emerged as a paradigm for decomposing complex tasks across specialized agents and coordinating their outputs (Wang et al., 2024; Yao et al., 2022; Li et al., 2023). Recent work, particularly with foundation models, has explored role specialization, inter-agent communication, planning, and memory(Fioresi et al., 2026; Zhang et al., 2026). Such coordinated multi-agent reasoning has been applied across domains including medical reasoning, video understanding, and other multimodal tasks(Kim et al., 2024; Zhang et al., 2024). Parallel developments in multi-agent reinforcement learning have investigated coordination and specialization among multiple agents(Wang et al., 2025). To the best of our knowledge, no prior work has applied multi-agent frameworks to game engine optimization. We bring these directions together by formulating engine optimization across heterogeneous multi-game platforms as semantic-aware policy prediction and introduce the SAMA-GEO benchmark to enable its study.

## 3 SAMA-GEO Dataset

The SAMA-GEO dataset consists of 5,193 Roblox games from 17 genres and 59 genre-subgenre pairs. Each sample game consists of its name, description, genre-subgenre information, additional metadata (maximum number of players, total number of platform visits, voice chat support availability and whether private servers are allowed for the game), along with 3 thumbnail images. Each game is annotated with 10 semantic labels and 3 policy labels. Fig. 1 shows a sample games from different genres. More visualizations from the dataset are shown in Supp. Section G.

### 3.1 Data Collection and Processing

Roblox provides a uniquely suitable setting for studying cross-game engine optimization. Its engine hosts thousands of games spanning a wide range of genres, gameplay styles, and system requirements, all running on a shared infrastructure with configurable policy parameters. For our dataset, game metadata, descriptions, genre-subgenre information, and thumbnail images are collected via publicly available Roblox APIs (details in Supp. Section C).

We sample games by popularity. However, popularity-based sampling alone risks over-representing dominant genres. Simulation games constitute the largest genre class on Roblox by a significant margin, and uniform sampling would produce a dataset skewed toward a small number of genres, undermining the cross-game generalization we aim to study. We instead follow a stratified monotonic tiered sampling strategy, binning each genre-subgenre pair by the number of available games

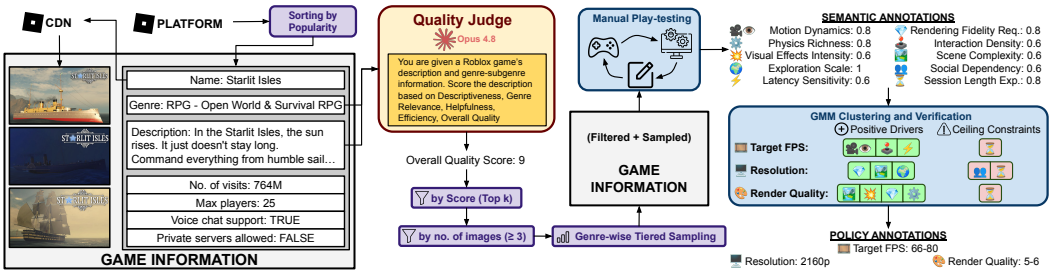



Figure 2: **Overview of the SAMA-GEO data flow pipeline.** The most popular Roblox games are sampled from the platform and images are obtained from Roblox's Content Delivery Network(CDN). A Quality Judge rates each game's description, and games are filtered based on score and minimum image count, followed by genre-wise monotonic tiered sampling to ensure fair representation. Games are manually play-tested by expert annotators to produce 10 semantic attribute annotations, which are translated to 3 policy labels via GMM-based clustering that captures multimodal structure in the configuration space, with positive drivers determining policy levels and ceiling constraints bounding maximum values. Resulting policy annotations are then verified by the annotators to ensure consistency and adherence to game-specific requirements.

and capping representation at each tier, while ensuring that no smaller genre class receives more representation than a larger one (Supp. Section B).

A further challenge is description quality. Game descriptions vary widely on Roblox, some are detailed and informative, while others are sparse, promotional, or uninformative. Since descriptions form a primary input to both annotation and model inference, low-quality descriptions introduce noise at every stage of the pipeline. We filter games using Claude Opus 4.8(Anthropic, 2026c) as a Quality Judge, scoring each description across five dimensions: descriptiveness, genre relevance, helpfulness, efficiency, and overall quality. Only games exceeding a quality threshold and having at least 3 thumbnail images are retained in the final dataset. Fig. 2 shows the entire data flow pipeline.

### 3.2 Annotation Pipeline

**Manual play-testing.** The central annotation challenge in SAMA-GEO is that optimal engine policies are not directly observable, they must be inferred from a deep understanding of how each game looks, feels, and plays. This rules out automated annotation, as surface-level features cannot capture nuanced gameplay characteristics that drive engine requirements. Expert annotators therefore play each gold set game multiple times across sessions before annotating, ensuring that semantic labels reflect genuine gameplay experience rather than superficial impressions. Each game is annotated on 10 semantic attributes on a scale of 0-1, capturing gameplay characteristics and 3 policy attributes governing engine configuration, summarized in Table 1(detailed definitions in Supp. Sections D, E).

**Semantic-to-Policy translation.** Translating semantic annotations to policy labels presents its own challenge. A game's optimal engine policy is not uniquely determined. Multiple FPS targets or render quality settings may provide comparable player experience, reflecting genuine trade-offs between performance and visual fidelity. We represent Target FPS and Render Quality as ranges of plausible configurations using GMM-based(Reynolds et al., 2009) clustering. Each policy is driven by a weighted composite of its most relevant semantic attributes, its drivers, while ceiling constraints cap the maximum policy level based on attributes that bound resource requirements. The GMM learns policy archetypes from gold annotations and assigns each game a range bracketing its estimated requirement. Resolution is retained as a single ground-truth value due to the discrete nature of its driver space. Each policy annotation is verified by annotators for consistency within genres and adherence to game-specific requirements. Full details are provided in Supp. Section F.

### 3.3 Gold and Silver splits

Manual play-testing produces high-quality annotations but is expensive, limiting its application to a subset of games. The dataset is therefore divided into two tiers: a gold set of 152 manually play-tested and expert-annotated games, and a silver set of 5,041 games annotated using GPT 5.6 Sol(OpenAI, 2026c) using gold annotations of the same genre as reference, grounding silver annotations in verified human judgments. Silver annotations undergo random verification by the annotators for consistency. The gold and silver sets follow 6-tiered and 11-tiered monotonic sampling respectively(details in Supp. Section B).

Table 1: **Summary of the 10 semantic and 3 policy annotation labels in SAMA-GEO.**

| Type | Attribute | Description |
|---|---|---|
| Semantic | Motion Dynamics | Pace of camera movement and player locomotion |
| | Rendering Fidelity Requirement | Importance of detail recognizability at distance |
| | Physics Richness | Extent to which physics simulation influences gameplay |
| | Interaction Density | Frequency of meaningful player actions during gameplay |
| | Visual Effects Intensity | Frequency and prominence of particles, lighting, and effects |
| | Scene Complexity | Visual density and structural detail of typical scenes |
| | Exploration Scale | Spatial openness and extent of the explorable game world |
| | Social Dependency | Reliance on multiplayer interaction for core gameplay |
| | Latency Sensitivity | Degree to which input delay disrupts gameplay |
| | Session Length Expectation | Typical duration of an uninterrupted gameplay session |
| Policy | Target FPS | Target frame rate the engine aims to maintain (15–120 fps) |
| | Resolution | Display resolution of the game (360p–2160p) |
| | Render Quality | Overall visual rendering complexity (levels 1–8) |

# 4 Multi-agent Framework

## 4.1 Problem Statement and Framework Overview

Given game-specific data as input, our objective is to predict an optimal set of engine policy parameters {target FPS, resolution, render quality}, that maximizes the player's experience for each game. Rather than predicting policy directly from game information, we establish a game's semantic profile as an explicit intermediate signal that grounds and informs policy prediction. This design is motivated by the heterogeneous nature of game semantics. Different aspects of a game's semantic profile drive different policy dimensions, and making this structure explicit enables more interpretable and game-specific policy prediction. We approach this with a multi-agent framework where specialized agents decompose semantic and policy reasoning into complementary roles, motivated by the diversity of information sources available, description, images, genre information, and metadata, and the distinct ways each source informs different aspects of the semantic profile and engine policy.

## 4.2 Constituent Agents

**Game Analyst.** The Game Analyst is the entry point of the framework, responsible for establishing the semantic profile of a game from its name, description, genre-subgenre information, and metadata. It jointly predicts semantic attributes and a base policy, as simultaneous prediction grounds the semantic profile toward policy-relevant characteristics, improving the quality of both. Downstream agents then operate on this semantic profile as an interpretable intermediate representation.

**Resource Aware Semantic Coordination (RASC) Refiner.** The RASC Refiner coordinates semantic information to recommend engine resource configurations, refining the base policy predictions using the semantic profile as explicit signals for each policy dimension. Rather than predicting policy from scratch, it treats the Game Analyst's base prediction as a strong prior and adjusts only where the semantic profile warrants it, preserving consistency while correcting semantic misalignments.

**Genre Analysis Agent.** Genre of a game is a strong driver in determining how it runs and how responsive it needs to be. Generally competitive, real-time PvP, and multiplayer games are fast-paced and need instant feedback to the player, while casual, turn-based and single player games are slower. Thus Genre Analysis agent further improves the target FPS prediction by analyzing the genre information, semantic profile and the previous target FPS prediction.

**Image Analysis Agent.** Resolution is the most visually impactful attribute for the player's experience. Therefore, images can provide the best visual perception of a game which in turn help the framework in determining the most suitable resolution. The Image Analysis agent further improves the resolution prediction by analyzing the images associated with each game, semantic profile and the previous resolution prediction.

At inference time, the Game Analyst processes a game's name, description, genre-subgenre information, and metadata to produce a semantic profile and base policy. The RASC Refiner refines the policy using the semantic profile, with its render quality prediction considered final. Target FPS and resolution are further refined by the Genre Analysis and Image Analysis agents respectively, each incorporating complementary information sources. The SAMA-GEO framework architecture is illustrated in Fig. 3, and the system prompts used for each agent are provided in Supp. Section H. The component-wise construction and analysis of the framework is detailed in Section 6.

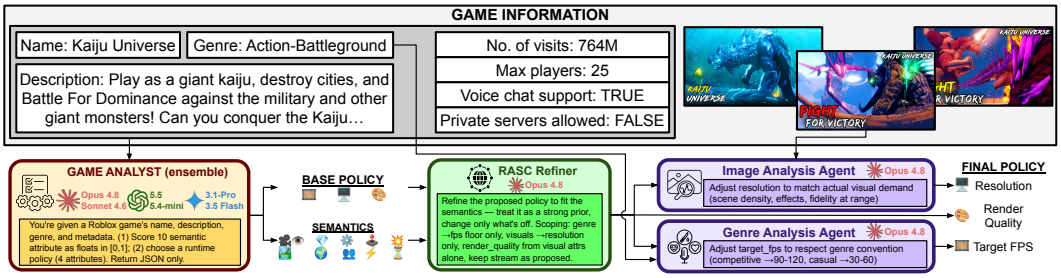



Figure 3: **Schematic Illustration of SAMA-GEO Framework Architecture.** Game information (except images) is processed by the Game Analyst ensemble to produce a semantic profile and base policy. The RASC Refiner refines the policy using the semantic profile as an explicit signal. The Genre Analysis Agent and Image Analysis Agent further refine target FPS and resolution respectively using genre conventions and visual evidence from thumbnail images, producing final policy.

## 5 Evaluation Suite

Game engine optimization across heterogeneous platforms introduces evaluation challenges that standard regression metrics alone cannot address. A model that predicts the dataset mean for every game may achieve acceptable pointwise error while producing identical policies for games with vastly different requirements, which standard metrics would not penalize. Similarly, a model could satisfy target policy constraints while assigning semantically incoherent configurations, or fail to reproduce the diversity of policies within a genre. We therefore introduce a dedicated eight-metric evaluation suite, with four metrics each for semantic and policy prediction. Of the eight metrics, two (MAE, RMSE) are established, five (SSP, MRD, CVR, PRS, GJSD) are adapted from existing metrics to suit our setting, and one (SPCS) is newly proposed.

### 5.1 Semantic Metrics

We evaluate semantic performance along three complementary dimensions: pointwise accuracy(MAE and RMSE), structural preservation(SSP), and attribute-wise spread(MRD).

**Mean Absolute Error(MAE).** MAE measures the average absolute difference between predicted and ground-truth values and is measured as

$$\mathrm{MAE} = \frac{1}{ND} \sum_{a=0}^{N-1} \sum_{i=0}^{D-1} \left| s_{a,i}^{\mathrm{gt}} - s_{a,i}^{\mathrm{pred}} \right| \tag{1}$$

where, $N$ is the number of samples(games), $D$ is the number of semantic attributes, and $s_{a,i}^{gt}$ and $s_{a,i}^{pred}$ are the ground-truth and predicted attribute values respectively.

**Root Mean Squared Error(RMSE).** RMSE measures the square root of the average squared difference between predicted and ground-truth values and is measured as

$$\mathrm{RMSE} = \frac{1}{N} \sum_{a=0}^{N-1} \sqrt{\frac{1}{D} \sum_{i=0}^{D-1} \left( s_{a,i}^{\mathrm{gt}} - s_{a,i}^{\mathrm{pred}} \right)^2} \tag{2}$$

where notation follows Eq. 1. MAE reflects typical prediction error, while RMSE tends to be more sensitive to larger deviations or outliers.

**Semantic Structure Preservation Score (SSP).** Both error metrics MAE and RMSE do NOT catch mode collapse. A model that predicts the dataset mean for every game can post a decent MAE while making all games identical (e.g. treating close-range and sniping FPS games the same). To capture this structural information, we introduce Semantic Structure Preservation (SSP) score, based on Spearman rank correlation(Spearman, 1904). SSP is calculated by first computing the full set of pairwise Euclidean distances among all games in ground truth semantic space, and again in predicted semantic space. All pairwise distances are then ranked in the ground-truth and predicted spaces and SSP is computed as their Spearman Rank Correlation.

$$d_{a,b}^{\mathrm{x}} = \sqrt{\sum_{i=0}^{D-1} \left( s_{a,i}^{\mathrm{x}} - s_{b,i}^{\mathrm{x}} \right)^2}; \mathrm{SSP} = 1 - \frac{6 \sum_{0 \le a < b \le N-1} \left( \mathrm{rank}\left( d_{a,b}^{\mathrm{gt}} \right) - \mathrm{rank}\left( d_{a,b}^{\mathrm{pred}} \right) \right)^2}{\frac{N(N-1)}{2} \left( \left( \frac{N(N-1)}{2} \right)^2 - 1 \right)} \tag{3}$$

where, $rank(d^x_{a,b})$ is the rank of the distance vector between games $a$ and $b$ in $x$-space. Higher SSP indicates stronger preservation of the relative semantic geometry, while lower values indicate structural distortion or representation collapse.

**Mean Range Difference(MRD).** We use Mean Range Difference (MRD)(Gaharwar et al., 2023), adapted from signal-reconstruction and trajectory evaluation paradigm, to multi-dimensional semantic representations(Kulkarni et al., 2026). For each semantic dimension, MRD compares the predicted and ground-truth dynamic ranges, computed over the same set of valid-prediction games

$$\mathrm{MRD} = \frac{1}{D}\sum_{i=0}^{D-1}\left|\left(\max_{0\le a\le N-1} s^{\text{pred}}_{a,i} - \min_{0\le a\le N-1} s^{\text{pred}}_{a,i}\right) - \left(\max_{0\le a\le N-1} s^{\text{gt}}_{a,i} - \min_{0\le a\le N-1} s^{\text{gt}}_{a,i}\right)\right| \quad (4)$$

Where $max$ and $min$ are respectively the maximum and minimum values for the attribute over a semantic space. Lower MRD indicates similar attribute-wise spread, while higher values indicate over- or under-dispersion. In particular, mode collapse produces a near-zero predicted range and therefore a large MRD. Thus, SSP evaluates whether the semantic structure is preserved, whereas MRD evaluates whether its spread is preserved.

### 5.2 Policy Metrics

For evaluating policy predictions, we focus on measuring policy range satisfaction(CVR and PRS), consistency with semantic profile(SPCS) and intra-genre diversity preservation(GJSD).

**Constraint Violation Rate(CVR)**. Because multiple policies may satisfy a game's target constraints, we evaluate constraint adherence rather than exact policy matching. We adapt Constraint Violation Rate to measure the proportion of policy attributes that violate their corresponding constraints(target FPS, render quality outside ground truth range, and resolution not in matching tier):

$$\mathrm{CVR} = \frac{1}{3N}\sum_{a=0}^{N-1}\left(\mathbf{1}\left[f^{\text{pred}}_a \notin [f^{\text{lo}}_a, f^{\text{hi}}_a]\right] + \mathbf{1}\left[r^{\text{pred}}_a \neq r^{\text{gt}}_a\right] + \mathbf{1}\left[q^{\text{pred}}_a \notin [q^{\text{lo}}_a, q^{\text{hi}}_a]\right]\right) \quad (5)$$

where, $f$ is target FPS, $r$ is resolution and $q$ is the render quality, and $x^{pred}$ is the prediction for attribute $x$, $x^{lo}$ and $x^{hi}$ are lower and upper bounds of the ground truth range, while $x^{gt}$ is the ground truth tier. CVR measures constraint adherence. "0" indicates that all evaluated constraints are satisfied, while higher values indicate more frequent violations.

**Policy Retention Score(PRS).** CVR does not distinguish marginal from substantial violations. We therefore adapt the soft constraint-retention score to provide partial credit based on the degree of adherence. For ranged attributes, predictions within the target range receive a score of 1, decreasing linearly outside the range. Resolution receives a graded score based on absolute resolution difference(implementation details are provided in Supp. Section I):

$$\sigma(x, \mathrm{lo}, \mathrm{hi}) = \max\left(0,\ \min\left(1,\ 1 - \frac{\max(\mathrm{lo}-x,\ x-\mathrm{hi})}{\mathrm{hi}-\mathrm{lo}}\right)\right); \mathrm{PRS} = \frac{1}{3N}\sum_{a=0}^{N-1}\left(\sigma\left(f^{\text{pred}}_a, f^{\text{lo}}_a, , f^{\text{hi}}_a\right) + \left(1 - \frac{\left|V(r^{\text{pred}}_a) - V(r^{\text{gt}}_a)\right|}{1800}\right) + \sigma\left(q^{\text{pred}}_a, q^{\text{lo}}_a, q^{\text{hi}}_a\right)\right) \quad (6)$$

where $x$ is an attribute prediction. PRS is averaged across FPS, resolution, and render quality over all games and $V(.)$ is the conversion function to kilopixels. Thus, CVR checks if a constraint is violated, while PRS captures degree of adherence, making PRS sensitive to near-feasible policies.

**Semantic-Policy Consistency Score (SPCS).** CVR and PRS assess constraint adherence but do not determine whether policies appropriately differentiate games according to their semantic characteristics. A model could therefore satisfy constraints while producing semantically incoherent policies. We propose Semantic-Policy Consistency Score (SPCS) to measure whether policy differences across games follow semantic-policy relationships like the ground-truth policy constraints.

For each game and policy dimension, we compute a signed composite of the relevant predicted semantic attributes according to the semantic-to-policy relationships defined by the policy-generation process (Supp. Section I), and measure its pairwise concordance with the corresponding predicted policy values:

$$\mathrm{SPCS_x} = \frac{\sum_{0\le a<b\le N-1}\mathbf{1}\left[(c^x_a - c^x_b)\left(x^{pred}_a - x^{pred}_b\right) > 0\right]}{\sum_{0\le a<b\le N-1}\mathbf{1}\left[(c^x_a - c^x_b)\left(x^{pred}_a - x^{pred}_b\right) \neq 0\right]}; \mathrm{SPCS} = \frac{1}{3}\left(\mathrm{SPCS}_f + \mathrm{SPCS}_r + \mathrm{SPCS}_q\right) \quad (7)$$

Table 2: **Component-wise performance on the gold set.** The performance improves with each additional component in different metrics. The final framework performs the best across all metrics.

| Configuration | MAE ↓ | RMSE ↓ | SSP ↑ | MRD ↓ | CVR ↓ | PRS ↑ | SPCS ↑ | GJSD ↓ |
|---|---|---|---|---|---|---|---|---|
| Naive (const) | 0.179 | 0.207 | NaN | 0.800 | 0.752 | 0.410 | NaN | 0.243 |
| Naive (random) | 0.181 | 0.233 | 0.012 | 0.600 | 0.739 | 0.461 | 0.540 | 0.165 |
| Genre only | 0.147 | 0.180 | 0.540 | 0.090 | - | - | - | - |
| + Description | 0.125 | 0.156 | 0.667 | 0.105 | - | - | - | - |
| + Metadata | 0.126 | 0.158 | 0.670 | 0.090 | - | - | - | - |
| + Policy-aux | 0.118 | 0.149 | 0.667 | 0.108 | - | - | - | - |
| 6-model ensemble (Game Analyst) | 0.113 | 0.140 | 0.695 | 0.088 | 0.653 | 0.520 | 0.802 | 0.131 |
| +RASC Refiner | 0.113 | 0.140 | 0.695 | 0.088 | 0.614 | 0.569 | 0.856 | 0.127 |
| + Genre Analysis Agent | 0.113 | 0.140 | 0.695 | 0.088 | 0.605 | 0.574 | 0.855 | 0.115 |
| **+ Image Analysis Agent (SAMA-GEO)** | **0.113** | **0.140** | **0.695** | **0.088** | **0.603** | **0.576** | **0.877** | **0.114** |

where, $c_a^x$ is the composite for game $a$ and attribute $x$. A pair is concordant when the semantic composite and policy differences have the same sign; pairs involving a tie in either quantity are ignored. The overall SPCS is the mean across the three policy dimensions. SPCS is therefore a pairwise, threshold-free measure of semantic-policy alignment: a high score indicates that predicted policies vary consistently with the model's predicted semantic characteristics, complementing CVR and PRS by rewarding meaningful policy differentiation even when multiple policies satisfy the target constraints.

**Genre-wise Jensen-Shannon Divergence (GJSD).** CVR, PRS, and SPCS do not directly assess whether the model reproduces the distribution of policies within a genre. We therefore adapt Jensen-Shannon Divergence (JSD) (Lin, 1991) to compare predicted and reference policy distributions within each top-level genre:

$$\text{GJSD} = \frac{1}{3|\mathcal{G}|} \sum_{g\in\mathcal{G}} \sum_{d\in\{f,r,q\}} \left[ \frac{1}{2} \sum_{c\in[d^{\text{lo}},d^{\text{hi}}]} P_{g,d}[c] \log_2 \frac{2P_{g,d}[c]}{P_{g,d}[c]+Q_{g,d}[c]} + \frac{1}{2} \sum_{c\in[d^{\text{lo}},d^{\text{hi}}]} Q_{g,d}[c] \log_2 \frac{2Q_{g,d}[c]}{P_{g,d}[c]+Q_{g,d}[c]} \right] \tag{8}$$

where, $P_{g,d}[c]$ and $Q_{g,d}[c]$ are the predicted and ground truth distributions of genre $g$, attribute $d$ for bucket $c$. The predicted distribution is formed from predicted values, while the reference distribution represents each game's target range across the categories it spans. The policy-tier definitions, discretization, and smoothing details are provided in Supp. Section I. GJSD is reported per genre with its sample count, with the overall score given by the unweighted mean across genres($\mathcal{G}$). Lower GJSD shows closer distributional agreement and better preservation of within-genre policy diversity.

## 6 Semantic-Aware Policy Prediction: A Component-wise Analysis

We validate the SAMA-GEO framework through a bottom-up component-wise analysis, examining the contribution of each component to semantic and policy prediction. We compare against two naive baselines: a constant baseline, a relatively general practice in adaptive engine platforms, assigning all games the same configuration of 60 FPS, 1080p, and render quality 3, reflecting a typical average Android device(as Android constitutes the largest user base on Roblox), and a random baseline. This analysis is performed on the gold set and results are reported in Table 2.

**Semantic Prediction.** We begin with genre-only input, which serves a dual purpose. It establishes whether genre alone is sufficient for semantic prediction, directly addressing the motivation for our semantic-aware formulation, and provides a natural starting point for the bottom-up analysis. While genre yields meaningful improvements over naive baselines, adding game description produces the single largest gain across MAE, RMSE and SSP, empirically confirming that substantial intra-genre variation exists and genre alone is insufficient for accurate semantic prediction. Metadata contributes to structural preservation and attribute-wise spread, as reflected by gains in SSP and MRD, while joint policy prediction further reduces error by aligning the semantic profile toward policy-relevant characteristics. To strengthen the Game Analyst, we ensemble predictions from six complementary backbones, namely, Claude Opus 4.8(Anthropic, 2026c), Claude Sonnet 4.6(Anthropic, 2026b), GPT 5.5(OpenAI, 2026b), GPT 5.4 mini(OpenAI, 2026a), Gemini 3.1 Pro(Google DeepMind, 2026a) and Gemini 3.5 Flash(Google DeepMind, 2026b), deliberately excluding frontier models such as Claude Fable 5(Anthropic, 2026a) and GPT 5.6 Sol and Terra(OpenAI, 2026c) to ensure a reproducible and fairly comparable reference point, as Sol was a part of the silver set annotation process(Section 3.3). The ensemble yields consistent gains across all four semantic metrics, though the framework is flexible and any backbone(s) can be employed for the Game Analyst depending on performance requirements and cost constraints.

Table 3: **Component-wise performance on the silver set.** The performance pattern is consistent with gold set performance. The final framework performs the best across all metrics.

| Configuration | MAE ↓ | RMSE ↓ | SSP ↑ | MRD ↓ | CVR ↓ | PRS ↑ | SPCS ↑ | GJSD ↓ |
|---|---|---|---|---|---|---|---|---|
| Naive (const) | 0.178 | 0.206 | NaN | 0.800 | 0.731 | 0.423 | NaN | 0.4671 |
| Naive (random) | 0.178 | 0.228 | 0.001 | 0.600 | 0.688 | 0.497 | 0.496 | 0.262 |
| Genre only | 0.157 | 0.188 | 0.540 | 0.111 | - | - | - | - |
| Game Analyst | 0.123 | 0.146 | 0.768 | 0.110 | 0.695 | 0.490 | 0.847 | 0.240 |
| **SAMA-GEO** | **0.123** | **0.146** | **0.768** | **0.110** | **0.655** | **0.542** | **0.897** | **0.150** |

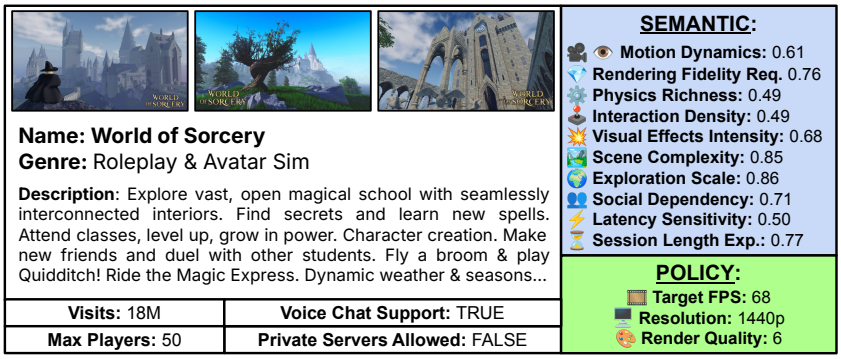



Figure 4: **Qualitative Example.** The game 'World of Sorcery' is a relatively popular game (18M visits) with a high player count capacity (50). The Game Analyst appropriately predicts moderate motion, interaction and latency, high scene complexity, world size and longer sessions. This semantic profile encourages the other Agents to determine a policy where the resolution and render quality is very high, while target FPS is average. This is compatible with an open-world magic based adventure game which needs to be visually appealing but does not have a high frame rate requirement.

**Policy Prediction.** With the semantic side of the framework established, we incorporate policy prediction agents sequentially(with base policy prediction of the Game Analyst serving as a starting point). The RASC Refiner shows broad improvement across CVR, PRS, SPCS and GJSD, demonstrating that semantic grounding improves policy prediction across all dimensions. The Genre Analysis Agent provides notable gains in CVR, PRS and GJSD, highlighting the impact of genre-aware reasoning on constraint adherence and distributional agreement. The Image Analysis Agent contributes most substantially to SPCS, demonstrating that visual evidence improves semantic-policy consistency. The complementary contributions of each agent across different metric dimensions justify their inclusion and confirm that semantic-aware multi-agent reasoning is effective for this problem. Backbone choice for the policy agents is discussed in Supp. Section A.

Fig. 4 illustrates a qualitative example on the SAMA-GEO framework for an open-world roleplay game, with additional examples across genres provided in Supp. Section I. We further evaluate on the silver set (Table 3), which at 5,041 games is substantially larger than the gold set. The consistent trend across both sets demonstrates that the framework generalizes well and is not biased toward any particular subset of the data.

## 7 Conclusion

We introduced SAMA-GEO, a benchmark and framework for studying semantic-aware policy prediction for game engine optimization across heterogeneous multi-game platforms. The SAMA-GEO benchmark comprises a two-tier dataset of 5,193 games with manually play-tested expert-annotated gold set and gold-referenced silver set, complemented by an eight-metric evaluation suite spanning semantic and policy prediction, including the newly proposed Semantic-Policy Consistency Score (SPCS) that captures semantic-policy alignment undetected by existing constraint-based metrics. We formulated game engine optimization as semantic-aware policy prediction, motivated by the insufficiency of genre alone for accurate policy prediction, empirically confirmed through our component-wise analysis, where genre-only input leaves substantial semantic information uncaptured. The SAMA-GEO framework, the first multi-agent system for this problem, decomposes semantic and policy prediction across specialized agents and demonstrates consistent and complementary gains as capabilities are incorporated. Together, these contributions provide an affirmative answer to our central question, semantic-aware multi-agent systems do work for game engine optimization, and establish a foundation for future research on generalizable engine policy prediction across diverse game workloads.

### AI USE STATEMENT

In this work, we used generative AI tools for implementing methods. LLM-generated code was verified and tested for correctness by the authors. We also use generative AI tools for partial annotation of a section(silver) of the dataset, using manual annotations of the other section(gold) as a reference. We have not used generative AI tools for generating complete data points, formulating mathematical claims, or any of the rest of the required disclosure tasks. Additionally, we used generative AI tools for refining the manuscript. We have reviewed all AI-assisted work. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

## REFERENCES


Anthropic. Claude fable 5 is rolling out. https://www.anthropic.com/news/claude-fable-5-is-rolling-out, July 2026a.

Anthropic. Introducing claude sonnet 4.6. https://www.anthropic.com/news/claude-sonnet-4-6, February 2026b. Accessed 2026-09-10.

Anthropic. Introducing claude opus 4.8. https://www.anthropic.com/news/claude-opus-4-8, May 2026c. Accessed 2026-09-10.

Thomas H Apperley. Genre and game studies: Toward a critical approach to video game genres. *Simulation & gaming*, 37(1):6–23, 2006.

Rachel Ivy Clarke, Jin Ha Lee, and Neils Clark. Why video game genres fail: A classificatory analysis. *Games and Culture*, 12(5):445–465, 2017.

Joseph Fioresi, Parth Parag Kulkarni, Ashmal Vayani, Song Wang, and Mubarak Shah. Learning to share: Selective memory for efficient parallel agentic systems. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=cCFyY2LmF5.

Anshuman Gaharwar, Parth Parag Kulkarni, Joshua Dickey, and Mubarak Shah. Xi-net: Transformer based seismic waveform reconstructor. In *2023 IEEE International Conference on Image Processing (ICIP)*, pp. 2725–2729. IEEE, 2023.

Google DeepMind. Gemini 3.1 pro. Model card, Google DeepMind, February 2026a. URL https://deepmind.google/models/model-cards/gemini-3-1-pro/.

Google DeepMind. Gemini 3.5 flash. Model card, Google DeepMind, May 2026b. URL https://deepmind.google/models/model-cards/gemini-3-5-flash/.

Jin Heo, Vic Wang, Ketan Bhardwaj, and Ada Gavrilovska. Stimpack: An adaptive rendering optimization system for scalable cloud gaming. In *23rd USENIX Symposium on Networked Systems Design and Implementation (NSDI 26)*, pp. 397–414, 2026.

Sirui Hong, Mingchen Zhuge, Jonathan Chen, Xiawu Zheng, Yuheng Cheng, Jinlin Wang, Ceyao Zhang, Steven Yau, Zijuan Lin, Liyang Zhou, et al. Metagpt: Meta programming for a multi-agent collaborative framework. In *International Conference on Learning Representations*, volume 2024, pp. 23247–23275, 2024.

Lanxiang Hu, Mingjia Huo, Yuxuan Zhang, Haoyang Yu, Eric P. Xing, Ion Stoica, Tajana Rosing, Haojian Jin, and Hao Zhang. lmgame-bench: How good are LLMs at playing games? In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=qeziG97WUZ.

Harold Jeffreys. An invariant form for the prior probability in estimation problems. *Proceedings of the Royal Society of London. Series A, Mathematical and physical sciences*, pp. 453–461, 1946.

Yuxiang Ji, Boyong He, Zhuoyue Tan, and Liaoni Wu. Game4loc: A uav geo-localization benchmark from game data. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 3913–3921, 2025.

Yuhang Jiang and Lukun Zheng. Deep learning for video game genre classification. *Multimedia Tools and Applications*, 82(14):21085–21099, 2023.

Niels Justesen, Philip Bontrager, Julian Togelius, and Sebastian Risi. Deep learning for video game playing. *IEEE Transactions on Games*, 12(1):1–20, 2019.

Yubin Kim, Chanwoo Park, Hyewon Jeong, Yik S Chan, Xuhai Xu, Daniel McDuff, Hyeonhoon Lee, Marzyeh Ghassemi, Cynthia Breazeal, and Hae W Park. Mdagents: An adaptive collaboration of llms for medical decision-making. *Advances in Neural Information Processing Systems*, 37:79410–79452, 2024.

Parth Parag Kulkarni, Rohit Gupta, Prakash Chandra Chhipa, and Mubarak Shah. Vidtag: Temporally aligned video to gps geolocalization with denoising sequence prediction at a global scale. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 23977–23987, June 2026.

Guohao Li, Hasan Hammoud, Hani Itani, Dmitrii Khizbullin, and Bernard Ghanem. Camel: Communicative agents for" mind" exploration of large language model society. *Advances in Neural Information Processing Systems*, 36:51991–52008, 2023.

Jianhua Lin. Divergence measures based on the shannon entropy. *IEEE Transactions on Information theory*, 37(1):145–151, 1991.

Prasanta Chandra Mahalanobis. On the generalized distance in statistics. *Sankhyā: The Indian Journal of Statistics, Series A (2008-)*, 80:S1–S7, 2018.

Ali Mohebali and Thiam Kian Chiew. Redefining game engine architecture through concurrency. In *International Conference on Intelligent Software Methodologies, Tools, and Techniques*, pp. 149–161. Springer, 2014.

OpenAI. Introducing gpt-5.4 mini and nano. `https://openai.com/index/introducing-gpt-5-4-mini-and-nano/`, March 2026a. Accessed 2026-09-10.

OpenAI. Gpt-5.5 system card. System card, OpenAI, April 2026b. URL `https://deploymentsafety.openai.com/gpt-5-5`.

OpenAI. Gpt-5.6: Frontier intelligence that scales with your ambition. `https://openai.com/index/gpt-5-6/`, July 2026c. GPT-5.6 family release, including GPT-5.6 Sol, Terra, and Luna.

Mingyu Ouyang, Siyuan Hu, Kevin Qinghong Lin, Hwee Tou Ng, and Mike Zheng Shou. Gameworld: Towards standardized and verifiable evaluation of multimodal game agents. *arXiv preprint arXiv:2604.07429*, 2026.

Douglas A Reynolds et al. Gaussian mixture models. *Encyclopedia of biometrics*, 741(659-663):3, 2009.

Anton Smerdov, Andrey Somov, Evgeny Burnaev, and Anton Stepanov. Ai-enabled prediction of video game player performance using the data from heterogeneous sensors. *Multimedia Tools and Applications*, 82(7):11021–11046, 2023.

C. Spearman. The proof and measurement of association between two things. *The American Journal of Psychology*, 15(1):72–101, 1904. ISSN 00029556. URL `http://www.jstor.org/stable/1412159`.

Juan J Vargas-Iglesias. Making sense of genre: The logic of video game genre organization. *Games and Culture*, 15(2):158–178, 2020.

Junlin Wang, Jue Wang, Ben Athiwaratkun, Ce Zhang, and James Zou. Mixture-of-agents enhances large language model capabilities. *arXiv preprint arXiv:2406.04692*, 2024.

Song Wang, Zhen Tan, Zihan Chen, Shuang Zhou, Tianlong Chen, and Jundong Li. Anymac: Cascading flexible multi-agent collaboration via next-agent prediction. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 11566–11578, 2025.

Qingyun Wu, Gagan Bansal, Jieyu Zhang, Yiran Wu, Beibin Li, Erkang Zhu, Li Jiang, Xiaoyun Zhang, Shaokun Zhang, Jiale Liu, et al. Autogen: Enabling next-gen llm applications via multi-agent conversation. *arXiv preprint arXiv:2308.08155*, 2023.

Yunfei Xie, Yinsong Ma, Shiyi Lan, Alan Yuille, Junfei Xiao, and Chen Wei. Play to generalize: Learning to reason through game play. In *International Conference on Learning Representations*, volume 2026, pp. 17213–17243, 2026.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik R Narasimhan, and Yuan Cao. React: Synergizing reasoning and acting in language models. In *The eleventh international conference on learning representations*, 2022.

Alex L Zhang, Thomas L Griffiths, Karthik R Narasimhan, and Ofir Press. Videogamebench: Can vision-language models complete popular video games? *arXiv preprint arXiv:2505.18134*, 2025.

Guibin Zhang, Muxin Fu, Kun Wang, Frank Wan, Miao Yu, and Shuicheng Yan. G-memory: Tracing hierarchical memory for multi-agent systems. *Advances in Neural Information Processing Systems*, 38:12988–13018, 2026.

Lu Zhang, Tiancheng Zhao, Heting Ying, Yibo Ma, and Kyusong Lee. OmAgent: A multi-modal agent framework for complex video understanding with task divide-and-conquer. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen (eds.), *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, November 2024.

Xiangxi Zheng, Linjie Li, Zhengyuan Yang, Ping Yu, Alex Jinpeng Wang, Rui Yan, Yuan Yao, and Lijuan Wang. V-mage: A game evaluation framework for assessing visual-centric capabilities in multimodal large language models. *arXiv e-prints*, pp. arXiv–2504, 2025.

## APPENDIX OVERVIEW

In this supplementary material, we provide additional insights on our dataset and method. We organize our supplementary into the following sections:

A. Choice of Backbone

B. Genre Distribution and Analysis

C. API Resources

D. Semantic Annotation Labels

E. Policy Annotation Labels

F. Semantic-to-Policy Annotation Translation

G. Additional Visualizations from the Dataset

H. System Prompts used for Agentic Calls

I. Important Notes Regarding the Evaluation Suite

J. Additional Qualitative Examples

## A CHOICE OF BACKBONE

In Section 6, we discuss the process of building our SAMA-GEO framework component-by-component and analyzing the contribution of each component in improving performance across 4 metrics evaluating semantic prediction and 4 metrics evaluating policy prediction. The Game Analyst agent in the framework primarily focuses on predicting semantic attributes and uses an ensemble backbone of 6 models, as discussed in Section 6. The framework is flexible in the way that it can use any number of backbones and depends on the user's performance requirements and cost constraints. The other 3 agents that focus on policy prediction, which are the RASC Refiner agent, the Genre Analysis Agent and the Image Analysis Agent use a single backbone common across all 3 agents for consistency.

In the main paper, we use Claude Opus 4.8(Anthropic, 2026c) as mentioned. Here, we perform an ablation to compare the performance of our framework across choices of backbone models for policy prediction. The results are reported in Table 4. We observe that inclusion of the 3 agents improves the performance of our framework regardless of the backbone used, as all variants are better than the 'Game Analyst only' version of the framework on all 4 policy metrics. Out of all the backbones, Claude Opus 4.8 yields the best results on CVR, PRS and SPCS and competitive results on GJSD. Hence, Claude Opus 4.8 was chosen for the final framework.

Table 4: **Ablation on choice of backbone for policy prediction.** Comparing variants of the framework with different backbones used in policy prediction agents. We show that inclusion of these agents works regardless of the choice of the backbone. Overall, Claude Opus 4.8 is the best choice.

| Family | Model | CVR ↓ | PRS ↑ | SPCS ↑ | GJSD ↓ |
|---|---|---|---|---|---|
| - | Game Analyst | 0.653 | 0.520 | 0.802 | 0.131 |
| Claude | Claude Opus 4.8(Anthropic, 2026c) | **0.603** | **0.576** | **0.877** | 0.114 |
| | Claude Sonnet 4.6(Anthropic, 2026b) | 0.638 | 0.545 | 0.825 | 0.112 |
| GPT | GPT 5.5(OpenAI, 2026b) | 0.612 | 0.570 | 0.868 | 0.125 |
| | GPT 5.4 mini(OpenAI, 2026a) | 0.616 | 0.563 | 0.857 | 0.121 |
| Gemini | Gemini 3.1 Pro(Google DeepMind, 2026a) | 0.653 | 0.539 | 0.850 | 0.121 |
| | Gemini 3.5 Flash(Google DeepMind, 2026b) | 0.636 | 0.551 | 0.810 | **0.110** |

## B GENRE DISTRIBUTION AND ANALYSIS

Genre forms a fundamental distinguishing factor among games, reflecting differences in gameplay mechanics, objectives, and player interactions that can influence the engine policies required by a

game. As our goal is to study engine policy prediction across multiple games, it is important that the dataset captures a diverse range of genres rather than being dominated by a particular type of game. Our dataset consists of games from Roblox from 17 different genres, many of which are further divided into subgenres. Subgenres are used especially for a more granular division of genres. Every game has a genre and optionally a subgenre giving us a total of 59 genre-subgenre pairs (including 'none' subgenre) for categorization. The 17 top-level genres are as follows:

1. **Action** - Action games emphasize active player participation and direct interaction with the game environment, often centered around challenges, combat, or other forms of moment-to-moment engagement. The genre encompasses a diverse range of gameplay experiences and can involve varying levels of complexity and player agency. Subgenres in this category are Battlegrounds & Fighting, Music & Rhythm, and Open World Action
2. **Adventure** - Adventure games typically emphasize exploration, discovery, and progression through game environments, often encouraging players to uncover content or complete objectives. They can range from open-ended exploration experiences to games driven by narrative, quests, or item discovery. Included subgenres are Exploration, Scavenger Hunt, and Story
3. **Education** - Educational games are designed to engage players in learning-oriented activities while maintaining interactive and game-based experiences. They may focus on acquiring knowledge, developing skills, or exploring concepts through structured gameplay and interactive challenges. There are no specific subgenres in this category
4. **Entertainment** - Entertainment games primarily focus on providing engaging and enjoyable experiences, with gameplay centered around interactive or media-oriented content. They can involve a range of experiences, from consuming and interacting with audiovisual content to exploring curated spaces and social hubs. Included subgenres: Music & Audio, Showcase & Hub, and Video
5. **Obby & Platformer** - Obby and platformer games focus on navigating environments through movement, timing, and spatial challenges, often requiring players to overcome obstacles to progress. These games can range from structured obstacle courses to more continuous platforming experiences centered around traversal and movement. Subgenres in this category are Classic Obby, Runner, and Tower Obby
6. **Party & Casual** - Party and casual games emphasize accessible, lightweight gameplay that can be enjoyed in short sessions or shared social settings. They often focus on simple interactions, playful challenges, and easily approachable experiences rather than complex or highly structured gameplay. Included subgenres are Childhood Game, Coloring & Drawing, Minigame, and Quiz
7. **Puzzle** - Puzzle games center on problem-solving and cognitive challenges, requiring players to reason through rules, patterns, or constraints to achieve a desired outcome. Gameplay typically emphasizes logical thinking, observation, and decision-making, with challenges varying in structure and complexity. Subgenres: Escape Room, Match & Merge, and Word
8. **Roleplay & Avatar Sim** - Roleplay and avatar simulation games focus on player-driven identity, social interaction, and the customization or simulation of characters and their activities. These games often provide open-ended experiences where players can shape their avatar, adopt different roles, and engage with virtual environments or social scenarios. Included subgenres are: Animal Sim, Dress Up, Life, Morph Roleplay, and Pet Care
9. **RPG** - Role-playing games center on assuming a character or role within a persistent game world, often involving progression, character development, and goal-oriented gameplay. They can vary substantially in how players interact with the world, from real-time combat and exploration to survival mechanics and turn-based encounters. Subgenres in this category are Action RPG, Open World & Survival RPG, and Turn-based RPG
10. **Shooter** - Shooter games focus on combat centered around ranged attacks, requiring players to engage targets through aiming, positioning, and tactical decision-making. Gameplay can involve competition against other players or encounters with computer-controlled opponents, with an emphasis on combat strategy and accuracy. Included subgenres: Deathmatch Shooter, and PvE Shooter

11. **Shopping** - Shopping games center on browsing, selecting, and acquiring virtual items, often incorporating elements of customization and collection. These experiences may focus on allowing players to personalize their virtual presence through the discovery and purchase of in-game items. Subgenres: Avatar Shopping
12. **Simulation** - Simulation games model or recreate systems, activities, or scenarios in which players interact with and influence the underlying game environment. The genre encompasses a particularly broad range of gameplay experiences, from managing resources and building systems to experimenting with physical interactions and operating vehicles. This is the largest genre category on the platform. Subgenres include Idle, Incremental Simulator, Physics Sim, Sandbox, Tycoon, and Vehicle Sim.
13. **Social** - Social games primarily emphasize player interaction, communication, and shared experiences within a virtual environment. Gameplay often centers on socializing, collaboration, or participation in activities with other players rather than pursuing a single structured objective. There are no specific subgenres in this category
14. **Sports & Racing** - Sports and racing games focus on competitive activities based on physical sports, driving, or other forms of competition. They typically emphasize skill, timing, and performance, with gameplay centered around competing against other players or game-controlled opponents. Subgenres: Racing, and Sports
15. **Strategy** - Strategy games emphasize planning, resource management, and decision-making, requiring players to consider different actions and their consequences to achieve specific objectives. Gameplay often involves tactical or strategic choices over time, with players balancing available resources, positioning, and competing goals. Included subgenres: Board & Card Games, and Tower Defense
16. **Survival** - Survival games focus on overcoming threats and challenges while managing the conditions necessary for continued progress or survival. Gameplay often involves adapting to hostile environments, avoiding or confronting dangers, and making decisions under resource or situational constraints. Subgenres in this category are 1 vs All, and Escape
17. **Utility & Other** - This category encompasses games and experiences that do not fit naturally into the other defined genres or are primarily oriented toward utility-based interactions. These experiences may include specialized, unconventional, or hybrid gameplay that does not align with a single established genre. There are no subgenres in this category.

Our dataset features games from all genre-subgenres pairs with representation based on the number of games of that genre-subgenre category on the platform. As discussed in Main paper Section 3.1, we sample the games based on a tiered-system to ensure a fair and monotonic representation. Figs. 5 and 6 show the distribution of games across genres (and subgenres) for the gold and silver set respectively. The gold set follows a 6-tiered representation and the silver set follows an 11-tiered system (Main paper Section 3.3). Tiered representation sampling works by binning each genre-subgenre pair (genre class) into one of 6 buckets (gold set) depending on the number of games in the genre class. We also ensure that the sampling is monotonic such that in cases near the boundary of two buckets, the smallest class in the higher bin always gets more representation in the dataset than the largest class in the lower bin. As the Silver set is much larger than the Gold set, we use an 11-tiered representation sampling strategy here. The monotonicity is more relevant here, as their is a higher chance of the bucket boundary discrepancy mentioned above, which we strategically prevent.

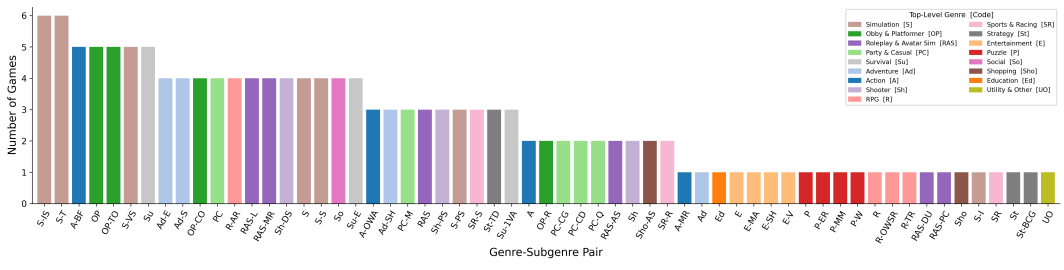



Figure 5: **Distribution of games across genres and subgenres in the gold set.** Columns representing subgenres within the same genre have the same color and each subgenre is assigned a code.

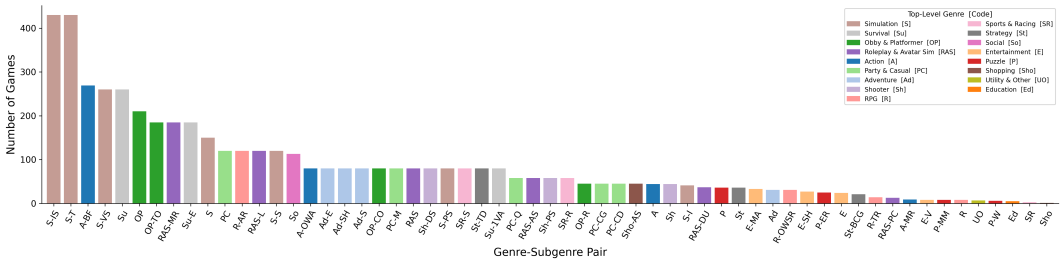



Figure 6: **Distribution of games across genres and subgenres in the silver set.** Columns representing subgenres within the same genre have the same color and each subgenre is assigned a code.

## C API Resources

These are the APIs we use for fetching game information from the Roblox platform and Roblox CDN. All APIs are available for public use.

- Games List API - for fetching universe_ids for top k games sorted by popularity
- Games API - for getting name, genre, subgenre, description, max players and number of visits for a particular game(universe_id)
- Games Media API - for fetching relevant image_ids for each game(universe_id)
- OpenCloud Universes API - for fetching availability details for voice chat support and private servers for a game(universe_id)
- Thumbnails API - for fetching images associated to a particular image_id from the CDN

## D Semantic Annotation Labels

As highlighted in main paper Section 3.2, semantic attributes are high-level, interpretable properties that characterize a game's underlying mechanics, structure, and gameplay dynamics. They provide a compact conceptual representation of a game, capturing key aspects of how it is designed and played. These attributes form the semantic anchor for the policy attributes, which describe how agents interact with and operate within the game. All semantic attributes are given a value between 0 and 1. In this section, we discuss each semantic attribute in detail and describe the aspects of the game it is intended to capture.

**Motion Dynamics.** This attribute captures the pace at which the player's viewpoint and focus change during gameplay, encompassing both camera movement and player locomotion. Lower values correspond to games with relatively static or slow-changing viewpoints, while higher values indicate increasingly rapid and continuous movement. A static game like chess will have a value close to 0, while a high pace arena first person shooter or a racing game will have a value close to 1.

**Rendering Fidelity Requirement.** This attribute captures the importance of maintaining visual detail and recognizability of gameplay-relevant objects across different viewing distances. It reflects whether gameplay primarily depends on detailed representations of nearby objects or requires distant objects to remain sufficiently detailed and identifiable. Higher requirements arise in games where long-range visual information is important to gameplay. A simple puzzle game might be on the lower end of the spectrum while a sniping focused shooter or a flight simulator would lean towards very high values.

**Physics Richness.** This attribute captures the extent to which physics simulation influences gameplay. It ranges from games where physics has little or no meaningful role to games where complex physical interactions are central to the gameplay experience. Higher physics richness reflects greater dependence on physical interactions, dynamics, and simulation as part of the game's mechanics. A simple game like chess where impact of physics is minimal will score near a 0, while a sandbox simulator, or games with physics as a core mechanic will have values near 1.

**Interaction Density.** This attribute captures how frequently meaningful player actions occur during gameplay. It reflects the overall pace of player decision-making and input, distinguishing games

with infrequent, deliberate actions from those requiring frequent or nearly continuous interaction. Higher interaction density indicates a greater frequency of meaningful actions that directly influence gameplay. For example a walking simulator needs minimal interactions from the player side and a first person shooter will need extremely frequent interactions from split second decisions.

**Visual Effects Intensity.** This attribute captures the frequency and prominence of visual effects during gameplay, including explosions, particles, bloom, dynamic lighting, smoke, and weather effects. It reflects the extent to which such effects contribute to the visual activity and presentation of the game. Higher intensity corresponds to gameplay environments where visual effects occur frequently or continuously. A board game will not have a large focus on visual effects while battle simulators and weather simulators would have a constant requirement for them.

**Scene Complexity.** This attribute captures the visual density and structural complexity of a typical game scene, considering the number, variety, and level of detail of visible objects and environmental elements. It reflects whether scenes are relatively sparse and simple or contain a large number of detailed and visually interacting elements. Higher scene complexity corresponds to increasingly dense and visually cluttered environments. A regular obby/platformer might not require very complex scenes, where as open-world cities would have a scene complexity close to 1.

**Exploration Scale.** This attribute captures the spatial extent and openness of the game world available for exploration. It reflects whether gameplay is confined to a small, self-contained space or spans multiple connected areas and expansive environments. Higher exploration scales correspond to increasingly large and open game worlds that support broader player movement and exploration. Simple avatar customization games are generally confined to a single room, while RPGs and massive open world adventure games have very large exploration scales.

**Social Dependency.** This attribute captures the extent to which gameplay depends on interaction or collaboration with other players. It distinguishes games that are primarily designed for individual play from those where multiplayer participation is an optional or integral component of the experience. Higher social dependency corresponds to games in which cooperative or player-to-player interaction plays an increasingly central role in gameplay. Solo experiences like PvE card games have a very low social dependency, whereas MMORPGs rely on the multiplayer experience for player enjoyment.

**Latency Sensitivity.** This attribute captures the extent to which gameplay depends on rapid and responsive input. It reflects how strongly delays between player input and game response affect the quality and playability of the experience. Higher latency sensitivity corresponds to games where even small increases in input delay can substantially disrupt gameplay. In franchise simulator games lag is barely noticeable, while fighting and first-person shooters rely on rapid movement and would be extremely sensitive to latency.

**Session Length Expectation.** This attribute captures the typical duration of an uninterrupted gameplay session. It reflects whether a game is generally designed around short, discrete play sessions or sustained periods of continuous play. Higher session length expectations correspond to games that typically encourage longer uninterrupted gameplay. Casual games tend to be short with sessions less than 10 minutes, while RPGs can be immersive resulting in long sessions of more than 60 minutes.

## E Policy Annotation Labels

In main paper Section 3.2, we state that policy attributes characterize the rules, parameters, and configuration choices that directly govern the behavior of the game engine. They determine how the game state evolves in response to player actions and environmental events, and therefore directly influence gameplay dynamics. In this section, we describe each policy attribute in detail and clarify the aspects of engine behavior it captures. These attributes constitute the primary targets for prediction in our game engine optimization framework.

**Target FPS.** This attribute specifies the target frame rate that the game engine should aim to maintain during gameplay. Since the achieved FPS is highly dynamic and can change frequently with the game state and workload, the target provides a stable reference for the engine to optimize toward rather than representing a fixed observed frame rate. A framework is therefore expected to predict the target FPS that the engine should aim to maintain. The target FPS is annotated as an integer

value between 15 and 120 frames per second. Franchise simulators running at a lower target FPS does not take away from the player experience, however for games like arena first-person shooters, target FPS should be as high as possible.

**Resolution.** This attribute specifies the display resolution at which a game is exhibited and is one of the strongest indicators of its visual presentation. It primarily determines the visual clarity and sharpness of the image, with higher resolutions providing a more clearly defined display output. For framework prediction purposes, we annotate the data with the p-notation ranging from 360p to 2160p(4K). However, it should be noted that game engines generally use resolution values in kilopixels which we actually use for computation of PRS as mentioned in main paper Section 5.2, and described in Supp. Section I.

**Render Quality.** This attribute captures the overall complexity and sophistication of the visual rendering requirements of a game. It encompasses factors such as rendering depth, shadows, anti-aliasing, visual effects, geometric detailing, reflections, ambient occlusion, and lighting complexity. Lower values correspond to simpler rendering requirements, while higher values indicate increasingly complex rendering pipelines with richer visual detail and effects. We annotate the data for this attribute on a scale of 1 to 8, where 1 refers to the most basic render devoid of any shadows, depth or reflections, while 8 is the most complex render with the highest quality of all the listed factors.

## F Semantic-to-Policy Annotation Translation

The semantic annotations described above (Section D) provide ten human-annotated attributes, each represented as a value in $[0, 1]$. We translate these semantic attributes into the ground-truth policy annotations (Section E) against which the agentic predictor is evaluated, using a GMM-based clustering framework (as mentioned in main paper Section 3.2) to capture the relationship between semantic characteristics and runtime policy requirements. The resulting policy annotations are device-agnostic representations of the computational and rendering demands of the game, rather than settings tied to a particular hardware configuration. Target FPS and render quality are represented as ranges, while resolution is represented as a categorical value; this formulation reflects the inherent variability in the mapping rather than imposing a closed-form deterministic relationship. In addition, the scoring procedure is stochastic, with annotations sampled independently on each pass, so that the predictor is evaluated on its ability to fall within a valid range rather than reproduce a single memorized value. The translation consists of two stages: fit, which learns the mapping from trusted gold annotations, and score, which applies the learned mapping to semantic annotations to produce the policy targets used for evaluation.

**The composite.** The core signal in the semantic-to-policy translation is a single scalar composite score computed independently for each policy attribute. For each policy, the relevant semantic driver attributes are normalized to $[0, 1]$, weighted according to the policy specification, summed, and rescaled to $[0, 1]$. Although the formulation supports signed weights in general, all drivers used by our three policies have equal positive weights, making the composite simply the average of the corresponding normalized semantic attributes. We also convert the composite into a continuous position on the policy tier ladder as composite $\times (L - 1)$, where $L$ is the number of policy levels; this position is subsequently used to map semantic demand to the corresponding policy values or ranges.

**Supporting helpers.** Several supporting operations are used to translate composite scores into policy annotations. Ceiling shift acts as a constraint on the maximum policy level by averaging the relevant ceiling attributes and applying a predefined threshold of 0.85; when the threshold is reached, the maximum level is reduced by one. Threshold binning maps a composite score to a discrete level by counting the number of thresholds it meets or exceeds, while level interpolation uses piecewise-linear interpolation to map fractional positions to values between adjacent policy levels. Finally, GMM responsibilities compute the soft membership of a game in each learned mixture component using the component weights, covariance matrices, and Mahalanobis distances (Mahalanobis, 2018) followed by a softmax. This responsibility computation is implemented directly in NumPy during scoring, allowing the fitted GMM to be used without requiring the clustering library at inference time.

**Policy Attributes.** Each policy attribute is determined by a distinct subset of semantic drivers. Target FPS is driven by latency sensitivity, motion dynamics, and interaction density, with session length acting as a ceiling constraint; its policy levels are anchored at 30, 45, 60, 90, and 120 FPS. Render quality is driven by scene complexity, visual effects intensity, rendering fidelity requirement, and physics richness, with session length providing the ceiling constraint; it uses discrete levels from 1 to 8. Resolution is driven by rendering fidelity requirement, scene complexity, and exploration scale, with session length and social dependency acting as ceiling constraints; it is represented categorically from 360p to 2160p. The selected drivers and their signs are based on domain knowledge, while all included drivers are assigned equal magnitude weights to avoid introducing unsupported assumptions about their relative importance.

**Learning the translation.** The fit phase is run once using only the trusted gold annotations. It identifies the driver and ceiling attributes for each policy, records input statistics, and reports any values outside the expected $[0, 1]$ range before fitting the policy-specific mappings. The three policies are then handled according to their output type. For resolution, no clustering is performed: the composite score is computed for every game and mapped to resolution tiers using explicit, domain-calibrated thresholds. In particular, 1080p serves as a broad default tier, while 1440p and 2160p are narrower premium tiers reserved for sufficiently high demand. Explicit thresholds are used because the three resolution drivers form a discrete five-by-five-by-five lattice, causing many games to share exactly the same composite value; quantile-based thresholds could therefore collapse a tier to zero games. This concentration of games at identical composite values also motivates the controlled scoring jitter described later, and both the selected thresholds and resulting gold-label distribution are recorded during fitting. For target FPS and render quality, the games' semantic driver vectors are used to fit full-covariance GMMs(Reynolds et al., 2009) with component counts ranging from 2 to 8, with 20 restarts and a fixed random seed for each candidate model. The model with the lowest BIC is selected, and a BIC-versus-component-count curve is recorded to characterize the dependence of model fit on the number of components. Each learned component represents a policy archetype and is mapped through the same composite formulation to a position on the corresponding policy ladder, retaining a fractional position for target FPS and rounding to a level index for render quality.

**Translation.** We tackle the translation for each attribute individually. For resolution, the game's composite score is perturbed with Gaussian jitter of standard deviation 0.03, mapped to a tier using the stored thresholds, adjusted by the ceiling shift, and finally clamped to the valid range. This jitter converts the repeated composite values arising from the discrete resolution driver lattice into controlled variation: games sharing the same composite can fall on different sides of a threshold across scoring passes, while the overall label distribution remains stable. For the ranged policies, the game's own composite is first converted to its position on the policy ladder, after which its soft GMM responsibilities are used to sample 400 archetype positions. The lower endpoint is the smaller of the game's own position and the 10th percentile of the sampled positions, while the upper endpoint is the larger of the game's own position and the 90th percentile; the upper endpoint is then adjusted by the ceiling shift, and both endpoints are clamped to the valid policy range. For target FPS, these positions remain fractional and are passed through the interpolation function to produce integer FPS values, allowing values between the predefined anchors; for render quality, the lower and upper positions are respectively floored and ceiled to level indices and mapped to the corresponding quality tiers. By combining the game's own demand with neighboring archetypes, this procedure generally produces a non-degenerate range that brackets the game's estimated policy requirement rather than assigning a single fixed value.

Thus, the GMM does one narrow job of discovering archetypes and soft memberships for the two ranged policies that have cluster structure while resolution, whose driver space is a degenerate lattice, is handled by threshold binning instead. Applying the model to the silver set changes only each game's own composite and membership; the archetypes and thresholds stay frozen from the gold fit.

## G Additional Visualizations from the Dataset

The dataset comprises 5,193 games from 17 genres(59 genre-subgenre pairs). To provide a quick peek into our dataset, we showcase extensive visualizations in the form of examples representing one game from each top-level genre with game name, genre-subgenre information, description, images and additional metadata of number of visits, maximum players in a session, availability of voice chat




Figure 7: **Extensive Visualizations of sample games from all top-level genres.** This figure shows one game from each of the 17 top-level genres, with images, genre info, descriptions and metadata.

support and the permission for private servers (Fig. 7). The visualizations are arranged in the same genre order as Section B for convenience.

# H SYSTEM PROMPTS USED FOR AGENTIC CALLS

**Game Analyst**

```
You are given a Roblox game's name, description, genre/subgenre, metadata. Produce two things:
  1. semantic — score each attribute as a float in [0.0, 1.0]:
    motion_dynamics: How rapidly does the player's viewpoint and focus change during gameplay — captures both camera movement and locomotion
(0 =  static/viewpoint never changes, 1 = frantic constant viewpoint shifts)
    scene_complexity: How crowded and visually detailed a typical scene is (0 = sparse/minimal, 1 = dense/highly detailed)
    rendering_fidelity_requirement: How important is it that distant gameplay-relevant objects remain visually detailed and identifiable (0 = only nearby
objects matter, 1 = very distant objects must be recognizable)
    visual_effects_intensity: Frequency of explosions, particles, bloom, lighting effects, smoke, weather (0 = none, 1 = constant heavy VFX)
    physics_richness: Importance of physics simulation — projectiles, destruction, ragdolls, vehicles (0 = none, 1 = physics-central)
    exploration_scale: World openness and encouragement to explore (0 = fully linear/arena, 1 = vast open world)
    latency_sensitivity: How much 100 ms of additional latency would hurt gameplay (0 = unnoticeable, 1 = game-breaking)
    social_dependency: Reliance on other players — co-op, PvP, social mechanics (0 = fully solo, 1 = requires many players)
    session_length_expectation: Typical uninterrupted session duration (0 = <2 min casual rounds, 1 = multi-hour persistent sessions)

  2. policy — choose a runtime policy:
    target_fps: integer 15-120
    resolution: one of "360p", "480p", "720p", "1080p", "1440p", "2160p"
    render_quality: integer, one of 1, 2, 3, 4, 5, 6, 7, 8 (low to high quality)

  Return ONLY valid JSON, no prose, no markdown:
{"semantic":
        {"motion_dynamics": <float 0-1>, "scene_complexity": <float 0-1>, "interaction_density": <float 0-1>, "rendering_fidelity_requirement": <float 0-1>,
        "visual_effects_intensity": <float  0-1>, "physics_richness": <float 0-1>, "exploration_scale": <float 0-1>, "latency_sensitivity": <float 0-1>,
        "social_dependency": <float 0-1>, "session_length_expectation": <float 0-1>},
  "policy":
        {"target_fps": <integer 15-120>, "resolution": <"360p"|"480p"|"720p"|"1080p"|"1440p"|"2160p">,"render_quality": <1|2|3|4|5|6|7|8>}
}
```

(a) **System Prompt for the Game Analyst Agent.**

**RASC Refiner**

```
You are the RASC — Resource Aware Semantic Coordinator — for the Roblox game engine.

You are given a game's semantic profile (10 normalized [0,1] attributes) and a PROPOSED runtime policy from an upstream analyst. Your job is to
REFINE the proposed policy so it better fits the game. Treat the proposal as a STRONG PRIOR — keep fields that are already appropriate and change
only what the semantics say is off. Do not rebuild from scratch.

How the semantics should shape each field:
target_fps (integer 15-120): primarily latency_sensitivity — competitive/fast > 0.7 → 90-120; moderate 0.4-0.7 → 60-90; casual/turn-based < 0.4 → 30-60;
fine-tune upward with motion_dynamics. Output the PRECISE integer the game needs — do NOT snap to 30/45/60/90.
resolution ("360p"|"480p"|"720p"|"1080p"|"1440p"|"2160p"): higher when scene_complexity, visual_effects_intensity, or rendering_fidelity_requirement
are high AND latency_sensitivity is low; lower when the game is performance-first (high motion + high latency_sensitivity).
render_quality (integer, one of 1,2,3,4,5,6,7,8): tier set by visual demand (VFX, scene complexity, fidelity). Output the canonical tier.

Return ONLY the refined policy as JSON, no prose, no markdown:
 {"target_fps": <int 15-120>, "resolution": <str>, "render_quality": <int>}
```

(b) **System Prompt for the RASC Refiner Agent.**

**Genre Analysis Agent**

```
You adjust a Roblox game's target_fps. Given the game's genre/subgenre, its semantic profile, and a PROPOSED target_fps, adjust the fps to respect
genre convention and responsiveness need: competitive/fast genres need a high floor (90-120); casual/turn-based sit lower (30-60). Treat the proposal
as a strong prior — change it only when the genre clearly warrants a different floor, and output the precise integer (do not snap to 30/45/60/90).

Return ONLY valid JSON, no prose: {"target_fps": <int 15-120>}
```

(c) **System Prompt for the Genre Analysis Agent.**

**Image Analysis Agent**

```
You adjust a Roblox game's resolution. Given the game's screenshots, its semantic profile, and a PROPOSED resolution, adjust it to match the actual
visual demand (scene density, effects, fidelity at range). Treat the proposal as a strong prior — change it only when the visuals clearly warrant a higher
or lower tier.

Return ONLY valid JSON, no prose: {"resolution": <"360p"|"480p"|"720p"|"1080p"|"1440p"|"2160p">}
```

(d) **System Prompt for the Image Analysis Agent.**

Figure 8: **System Prompts for each of the 4 agents in the SAMA-GEO framework.** These prompts ensure consistent and interpretable outputs during inference.

Fig. 8 shows the prompts that we use for each agent in the framework. Game Analyst (Fig. 8a) inputs game's name, description, genre/subgenre, metadata and outputs the semantic profile and a base policy prediction. RASC Refiner (Fig. 8b) inputs the semantic profile and the base policy prediction and refines the policy based on the semantics. Genre Analysis Agent (Fig. 8c) inputs the semantic profile and RASC refined policy along with genre-subgenre of the game and further

improves the Target FPS prediction. Similarly, Image Analysis Agent (Fig. 8d) inputs the images associated with the game and improves the resolution prediction within the final policy.

# I Important Notes Regarding the Evaluation Suite

**Policy Retention Score (PRS).** For resolution, PRS uses the absolute difference in vertical resolution rather than the difference between resolution tiers. The predicted and ground-truth resolution values are therefore converted from the p-notation used to specify resolution tiers to their corresponding values in kilopixels before computing the penalty($V(.)$ in Eq. 6). This ensures that larger resolution deviations receive proportionally larger penalties; for example, a prediction of 2160p for a 1440p target incurs a larger penalty than a prediction of 480p for a 360p target.

For ranged attributes such as FPS and render quality, predictions within the target range receive a score of 1, with the score decreasing linearly as the prediction moves outside the target range. The resulting scores are averaged across FPS, resolution, and render quality over all evaluated games.

**Semantic-Policy Consistency Score (SPCS).** SPCS uses the signed semantic-to-policy relationships inherited directly from the policy-generation process. These relationships define the semantic characteristics expected to correspond positively or negatively with each policy dimension and therefore determine the signed composite used in Eq. 7.

For FPS, the positively associated semantic attributes are Latency Sensitivity, Motion Dynamics, and Interaction Density. For resolution, the positively associated attributes are Rendering Fidelity Requirement, Scene Complexity, and Exploration Scale. For render quality, the positively associated attributes are Scene Complexity, Visual Effects Intensity, Rendering Fidelity Requirement, and Physics Richness.

For each game and policy dimension, these attributes are combined according to their corresponding directions to produce the semantic composite $c_a^x$. SPCS then evaluates the pairwise concordance between these composites and the corresponding predicted policy values. A pair is concordant when the semantic composite and policy differences have the same sign; pairs involving a tie in either quantity are ignored. The overall score is the mean across the three policy dimensions.

These relationships are directly inherited from the policy-generation process used to construct the ground-truth policy constraints, rather than being inferred from model predictions or selected based on evaluation results. This keeps SPCS aligned with the semantic-to-policy structure underlying the evaluation setting.

**Genre-wise Jensen-Shannon Divergence (GJSD).** GJSD compares predicted and reference policy distributions separately for each top-level genre and policy dimension. The predicted distribution is formed from a histogram of predicted policy values. The reference distribution represents the uncertainty inherent in the target policy ranges: when a target range spans multiple categories, probability mass is distributed uniformly across the categories covered by that range, while a single-category target is represented as a one-hot distribution.

For render quality, the corresponding quality tiers are used directly as the distribution categories. For FPS, values are discretized into the tiers 15, 30, 45, 60, 75, 90, 105, and 120 FPS, with range endpoints snapped to the nearest available tier. Thus, a target range spanning two render-quality tiers, such as tiers 16-17, assigns probability 0.5 to each tier.

Before normalization, both predicted and reference distributions are smoothed using Jeffreys smoothing with $\alpha = 0.5$(Jeffreys, 1946). This prevents zero-probability categories from producing degenerate distribution comparisons.

GJSD is computed separately for each genre-policy-dimension pair. We report GJSD per genre together with its sample count, and compute the overall GJSD as the unweighted mean across top-level genres. Lower GJSD indicates closer agreement between predicted and reference policy distributions and therefore better preservation of within-genre policy diversity.

## J Additional Qualitative Examples

In this section we show some additional qualitative examples. In main paper Section 6, we showed an example with an Open World Roleplay game. For that case, the resolution and render quality needed to be very high, while average target FPS was acceptable. Here, we show two more examples, one of an Arena first person shooter, where the reverse of the above case is desired, and the second of a Franchise simulator without any demanding requirements. Figs. 9 and 10 show that our framework predicts the semantic profile and policy accurately in both cases.

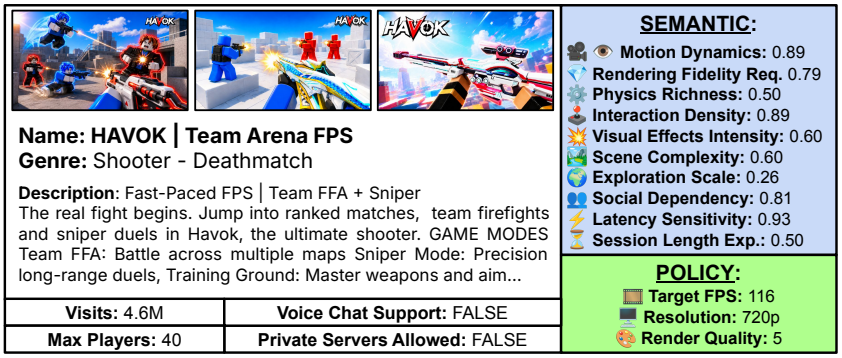



Figure 9: **Qualitative Example.** The game 'HAVOK — Team Arena FPS' is a popular game (4.6M visits) with a high player count capacity (40). The Game Analyst appropriately predicts very high motion, interaction and latency, and medium values for scene complexity, world size and average length sessions. This semantic profile encourages the other Agents to determine a policy where the target FPS is very high, while resolution and render quality is average. This is compatible with a fast-paced first person deathmatch shooter game which needs to have a very high frame rate as each split-second decision matters, while visual demand is not that high.

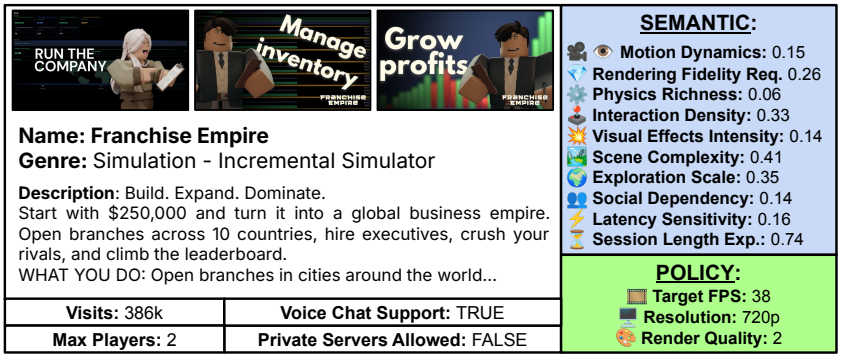



Figure 10: **Qualitative Example.** The game 'Franchise Empire' is a lower popularity game (386k visits) with a 2 player count capacity. The Game Analyst appropriately predicts low everything except for session length. This semantic profile encourages the other Agents to determine a policy where there is no specific attribute with a high resource demand. This is compatible with a franchise simulator which is neither visually demanding nor fast-paced which our framework correctly recognizes.