RefAV-CoFi: Coarse-to-Fine Program Ensembling with Multimodal Semantic Calibration for Scenario Mining

Yanchao Xu[1] Yufan Shu[2,1] Lian Liu[1]
[1]Guangzhou Automobile Group Co., Ltd
[2]School of Electronic Information and Electrical Engineering, Shanghai Jiao Tong University
xycbit2020@gmail.com syf574713782@outlook.com linger.ml@qq.com
Argoverse 2 Scenario Mining Challenge Technical Report

Abstract

We present RefAV-CoFi, a symbolic-program ensemble approach for the Argoverse 2 Scenario Mining Challenge. Scenario mining requires localizing natural-language descriptions in long autonomous-driving logs, and the RefAV benchmark frames this problem as planning-centric spatio-temporal retrieval over Argoverse 2 sensor data [1, 2]. A strong baseline translates each query into an executable program over tracked objects, but single-pass code generation is brittle: tracker noise, missing visual semantics, and unstable temporal thresholds can lead to both false positives and false negatives. We construct a bank of candidate programs from multiple LLM generations and prompting contexts, execute them as weak temporal miners, and combine their predictions with a prompt-family-aware selector. Scene-level prompts such as weather and lighting are calibrated with multimodal visual cues, while motion and interaction prompts use conservative multi-source temporal consensus. Our main empirical finding is that broad recall expansion and global smoothing often degrade balanced accuracy, whereas sparse high-confidence recall additions improve the Temporal Track score. Our best test variant improves the original test baseline from 65.47 to 68.97 Timestamp BA.
Keywords: Scenario Mining, Large Language Models, Program Ensembling, Multimodal Calibration, Autonomous Driving.

1 Introduction

Autonomous vehicles collect large volumes of synchronized camera, LiDAR, map, and tracking data. Mining rare or safety-critical scenarios from these logs is important for regression testing, data curation, and safety analysis. The RefAV benchmark [1] revisits scenario mining as a natural-language-driven spatio-temporal retrieval task built on Argoverse 2 [2]: given a driving log and a text prompt, a system must determine whether the described event occurs and localize the referred objects over time.

The RefAV program-synthesis baseline is attractive because it converts free-form language into executable code over a library of atomic geometric and kinematic functions [1]. This provides interpretability and allows complex prompts to be decomposed into object states, relations, and logical operators. However, our experiments revealed three persistent failure modes. First, individual generated programs are sensitive to wording and threshold choices. Second, upstream tracking outputs contain noise, especially for vulnerable road users and large vehicles. Third, several prompt families, such as weather, lighting, construction, and official-signal scenes, are weakly represented by tracker-only geometry.

RefAV-CoFi addresses these issues with a coarse-to-fine program ensemble. We generate multiple candidate programs, execute them as weak temporal miners, and then select only high-confidence timestamp additions over a stable base prediction. Compared with the SM-Agent solution, which emphasizes global context-aware generation and refiner agents [7], our final system places more emphasis on calibrated prediction aggregation: the program bank is treated as a set of noisy temporal hypotheses, and a prompt-family-aware selector decides when to trust each source.

Our contributions are:

1. We build a multi-source symbolic program bank from baseline, rerolled, and expert-context LLM code generations.
2. We introduce a prompt-family-aware temporal selector that uses sparse union, source agreement, and per-case frame caps rather than global smoothing or destructive replacement.
3. We integrate multimodal scene calibration for prompts whose semantics are not captured well by 3D tracks alone, such as weather and lighting.

2 Related Work

Scenario mining and RefAV. Early scenario mining systems often used structured queries, taxonomies, or handcrafted rules, which are interpretable but difficult to scale to compositional natural language. RefAV [1] introduced a planning-centric benchmark with 10,000 prompts over 1,000 Argoverse 2 logs and proposed referential tracking by program synthesis. The challenge solution SM-Agent [7] explored LLM agent architectures, including global context-aware generation and it-

erative refinement. More recent work argues for coarse-to-fine retrieval, combining visual semantic filtering, knowledge-base retrieval, and fine-grained matching [8, 9].

Program synthesis for visual reasoning. Visual programming methods such as VisProg [5] and ViperGPT [6] use language models to compose executable operations for visual reasoning. RefAV adapts this philosophy to dynamic 3D driving logs by defining atomic functions over object tracks. Our work follows the same symbolic-program view, but focuses on ensemble calibration after execution rather than only improving a single program.

Multimodal semantic filtering. Vision-language models such as CLIP [4] align images and text in a shared embedding space and have been used for retrieval and open-vocabulary perception. In scenario mining, multimodal semantics can serve as a coarse filter before fine-grained trajectory reasoning [8, 9]. We use this insight selectively: visual cues calibrate scene-level prompts, but they are not used as a global replacement for geometric reasoning.

3 Method

3.1 Baseline Symbolic Programs

For each log identifier l and natural-language prompt q, an LLM generates a Python program that calls RefAV atomic functions over tracker annotations. The program outputs timestamp-level predictions for referred objects. For the Temporal Track, we convert these outputs into the official referred-object timestamp format.

3.2 Program Bank

We build a bank of candidate programs from several sources: the original baseline code, Qwen-family rerolls, GPT-5.5 expert-context rerolls, GPT-5.5 natural-context rerolls, and selected train/validation-inspired repairs. Each program bank is executed on the test split. If a program fails or does not cover a case, we fall back to the stable baseline for that case. This makes each source a complete but noisy weak miner.

3.3 Multimodal Scene Calibration

Some prompts are not reliably solvable with tracked-object geometry alone. For weather and lighting prompts, visual labels or image-text semantic cues decide whether scene-level positives should be added or preserved. We use these signals as conservative gates rather than full replacements, because aggressive visual clearing was observed to hurt Timestamp BA.

3.4 Family-Aware Temporal Ensembling

Let $P_0(l, q, t)$ be the base binary prediction for log l, prompt q, and timestamp t. Let $P_i(l, q, t)$ denote the prediction from candidate program bank i. A naive union $\vee_i P_i(l, q, t)$ improves recall but also introduces many false positives. We therefore predict:

$$\hat{P}(l, q, t) = P_0(l, q, t) \vee A(l, q, t), \tag{1}$$

where $A(l, q, t)$ is a sparse addition set selected from the program bank. The selector uses the prompt family of q, source agreement among P_i, and a strict per-case frame cap. For scene families, A can be activated by visual confidence. For relation and motion families, A requires multi-source agreement or a very small frame budget. We avoid destructive replacement in the final system because replacement variants were sensitive to tracker noise and code-generation failures.

Table 1: Held-out test-server results for representative frozen variants. Timestamp BA is the primary target.

Variant	Main idea	Timestamp BA	Log BA
RefAV-style baseline	single program set	65.47	66.89
Scene calibration	weather/light visual gate	65.79	67.20
Program-bank selector	sparse multi-source additions	68.87	68.34
Scene + expert selector	visual and GPT-program micro additions	68.94	68.20
RefAV-CoFi final	source-agreed sparse selector	**68.97**	68.24

4 Experiments

4.1 Dataset and Metrics

We evaluate on the RefAV Scenario Mining benchmark [1], which is based on the Argoverse 2 Sensor dataset [2]. The benchmark contains 1,000 driving logs and 10,000 planning-centric natural-language prompts. We focus on the Temporal Track. Timestamp BA measures balanced binary accuracy at the timestamp level, while Log BA measures whether the scenario occurs at the log/prompt level. Although HOTA [3] is important for track-level evaluation, our submission strategy prioritizes Timestamp BA.

4.2 Submission Format

We use the official full-frame schema but retain only predicted REFERRED_OBJECT instances. This compact format preserves real geometry, scores, track identifiers, and labels, and a sanity submission verified identical Timestamp/Log scores for the Temporal Track. We do not claim full HOTA equivalence because non-referred objects are removed.

4.3 Main Results

Table 1 summarizes representative frozen variants rather than an exhaustive submission history. Development decisions were guided by training and validation diagnostics, including the reviewed validation failure modes in Fig. 2. The final system adds only a small number of high-confidence timestamps over a stable base, supporting the conclusion that the metric rewards precise temporal support more than broad positive expansion.

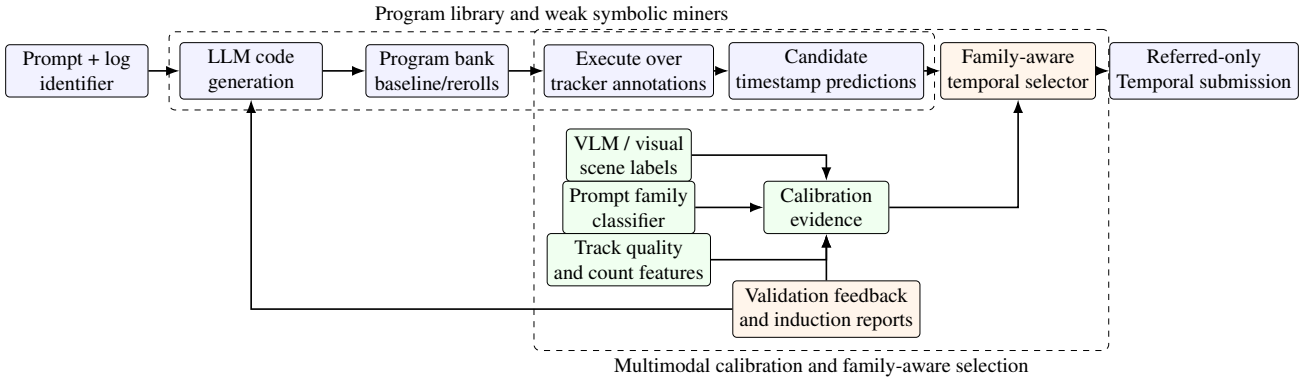

Figure 1: RefAV-CoFi overview. Multiple LLM-generated symbolic programs are executed as weak scenario miners. The selector combines candidate predictions with visual scene labels, prompt-family metadata, track-quality cues, and conservative temporal caps to produce a referred-only Temporal Track submission.

Table 2: Representative development ablations. Negative results were useful for shaping the final selector: broad recall and global post-processing were not reliable.

Variant	Intervention	Changed scale	Timestamp BA	Observation
Cleanup/smoothing	global temporal post-processing	broad temporal edits	65.04	Log improved, Timestamp dropped
Label-lite sanity	simplified temporal schema	format-level change	60.10	unsafe format for preserving scores
Core recall union	union of selected candidate banks	+450 frames	68.77	too broad for Timestamp
Reroll recall union	union with rerolled programs	+900 frames	68.72	broad reroll union hurts
Family union	family-level aggressive union	+1067 frames	68.71	still too many false positives
Extreme recall	stress-test recall expansion	+2404 frames	68.82	large Log degradation
Scene micro	visual-scene additions only	+13 frames	68.89	tiny scene recall helps
GPT micro	expert-program additions only	+37 frames	68.88	sparse GPT additions help modestly
Scene + GPT micro	combined sparse additions	+55 frames	68.94	strong sparse-recall baseline
Source-agreed final	two-source GPT agreement	+13 frames	**68.97**	best Timestamp so far

4.4 Ablation and Negative Results

Table 2 highlights a consistent pattern. Broad post-processing can increase the number of positive timestamps, but balanced accuracy penalizes the resulting false positives. Conversely, tiny additions supported by scene semantics or agreement among program banks improve recall without collapsing specificity. This is why the final selector uses sparse additions rather than a global union.

5 Discussion and Future Work

Engineering value. The method remains auditable because each candidate is an executable symbolic program, and each promoted change can be traced to a small set of log/prompt/timestamp additions. This makes it practical for competition debugging and for safety-analysis workflows where interpretability matters.

Why sparse recall works. Timestamp BA is sensitive to the temporal extent of predicted positives. Excessive smoothing and broad unions add false positive timestamps, while full replacement can remove useful baseline positives. Sparse additions provide a better trade-off: they target likely false negatives while preserving the stable baseline.

Future work. The current selector is rule-based. A more principled version should learn a selector from train/validation overlap using prompt embeddings, image-text similarity, model agreement, base positive count, track quality, and temporal span features. Repeated failures in turning, relation, and vulnerable-road-user prompts also motivate new atomic functions for robust turning detection, relative-motion reasoning, and tracker-noise modeling.

6 Conclusion

RefAV-CoFi improves RefAV-style scenario mining by turning LLM-generated programs into an ensemble of weak temporal miners and calibrating them with multimodal and prompt-family-specific signals. The final system emphasizes sparse high-confidence recall rather than broad post-processing, improving the original test baseline by more than three Timestamp BA points.

References

[1] C. Davidson, D. Ramanan, and N. Peri. RefAV: Towards planning-centric scenario mining. arXiv:2505.20981, 2025.

Figure 2: Reviewed validation failure modes that motivated conservative selection. Turning prompts can produce all-positive false positives when generated programs over-trust weak yaw cues. Vulnerable-road-user prompts are sensitive to tracker noise and relation thresholds, motivating track-quality diagnostics rather than global filtering.

[2] B. Wilson, W. Qi, T. Agarwal, J. Lambert, J. Singh, S. Khandelwal, B. Pan, R. Kumar, A. Hartnett, J. K. Pontes, D. Ramanan, P. Carr, and J. Hays. Argoverse 2: Next generation datasets for self-driving perception and forecasting. arXiv:2301.00493, 2023.

[3] J. Luiten, A. Osep, P. Dendorfer, P. Torr, A. Geiger, L. Leal-Taixe, and B. Leibe. HOTA: A higher order metric for evaluating multi-object tracking. International Journal of Computer Vision, 129:548–578, 2021.

[4] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, and I. Sutskever. Learning transferable visual models from natural language supervision. In ICML, 2021.

[5] T. Gupta and A. Kembhavi. Visual programming: Compositional visual reasoning without training. In CVPR, 2023.

[6] D. Suris, S. Menon, and C. Vondrick. ViperGPT: Visual inference via Python execution for reasoning. In ICCV, 2023.

[7] D. Chen, H. Zheng, W. Han, R. Tao, Z. Qiu, J. Yang, and J. Shen. SM-Agent solution for AV2 2025 Scenario Mining Challenge. WAD CVPR technical report, 2025.

[8] Y. Chen and R. Greer. Robust scenario mining assisted by multimodal semantics. In ICCV Workshop, 2025.

[9] Y. Chen and R. Greer. SMc2f: Robust scenario mining for robotic autonomy from coarse to fine. arXiv:2601.12010, 2026.