Driving scenario generation and evaluation using a structured layer representation and foundational models

Arthur Hubert, Gamal Elghazaly, Raphaël Frank
Interdisciplinary Centre for Security, Reliability and Trust (SnT), University of Luxembourg
29 Avenue John F. Kennedy, L-1855 Luxembourg, Luxembourg
firstname.lastname@uni.lu

***Abstract*—Rare and challenging driving scenarios are critical for autonomous vehicle development. Since they are difficult to encounter, simulating or generating them using generative models is a popular approach. Following previous efforts to structure driving scenario representations in a layer model, we propose a structured five-layer model to improve the evaluation and generation of rare scenarios. We use this model alongside large foundational models to generate new driving scenarios using a data augmentation strategy. Unlike previous representations, our structure introduces subclasses and characteristics for every agent of the scenario, allowing us to compare them using an embedding specific to our layer-model. We study and adapt two metrics to evaluate the relevance of a synthetic dataset in the context of a structured representation: the diversity score estimates how different the scenarios of a dataset are from one another, while the originality score calculates how similar a synthetic dataset is from a real reference set. This paper showcases both metrics in different generation setup, as well as a qualitative evaluation of synthetic videos generated from structured scenario descriptions. The code and extended results can be found at https://github.com/Valgiz/5LMSG.**

***Keywords*—Autonomous driving, Scenario engineering, Synthetic data, Large language model (LLM)**

I. Introduction

Edge Cases, Corner Cases, or Out of Distribution Scenarios are different names that generally refer to the most rare, challenging, or risky driving scenarios. They are very valuable for autonomous vehicle research, as they may be very challenging for learning-based systems [1], but they are also scarce among driving datasets by definition. Thus, a common workaround for researchers is to rely on simulations [2] and the generation of synthetic scenarios to acquire additional data [3, 4]. Since definitions and names can vary across the literature [5], this paper uses the term Edge Cases (ECs), to refer to rare and unexpected scenarios. They may consist of objects, agents, or behaviors that would not be expected to be found in a *normal* everyday situtation. We specifically chose not to focus on the risky aspect of safety-critical scenarios, but more on the rarity aspect. Despite this limitation, knowing if a scenario can be considered rare is subjective, which reduces the specificity of this definition. Thus, our work focuses on scenarios that are rare relative to a reference set of known scenarios. What makes a scenario an EC will thus depend on that reference.

Our goal is to generate synthetic visual data of driving scenarios using generative AI that rely on World Foundation Models (WFM) [6, 7]. However, the variety of possible scenarios is extremely high. Before we generate visual data, we first focus on creating the content of ECs. In order to create these scenario descriptions, we rely on the vast general knowledge of Large Language Models (LLM) by guiding their capabilities to generate new scenario ideas. Following previous work on LLM-based driving scenario generation [3, 4, 8], the challenge of this approach lies, not only in the proposed generation strategy, but also in the evaluation method and the scenario format. Both evaluation and structure depend on the use case and the type of scenario we aim to generate [4]. We propose a new format for textual representation of scenarios as a structured layered model [9], as well as an evaluation strategy tailored to this representation. Since physical realism can be challenging to measure from textual descriptions alone, we choose to generate synthetic scenarios based on existing real scenes to provide stronger chances that our scenarios will be coherent. Our generation strategy consists of augmenting a real dataset by editing targeted components of each scenario, and evaluating how our synthetic scenes differ from the original in order to grasp whether or not they constitute ECs. Rather than retraining a model with specific driving data, we study a prompting strategy on large pre-trained models. Our main contribution in this paper consists of the following:

1) Proposing a structured 5-layer model for driving scenario representation.
2) Proposing a new synthetic scenario generation strategy for driving dataset augmentation.
3) Studying new text-based evaluation methods for driving scenarios.

The rest of the paper is organized as follows. Section II covers related works in synthetic driving data generation. In Section III, we explain our scenario representation, our generation strategy, and our evaluation metrics. Finally, we discuss our experimental results in Section IV before concluding this paper in Section V.

II. Related works

A. Conditioned video generation

Conditional video generation has emerged as a prominent direction in visual generative modelling. Recent advances have shown substantial improvements both in visual fidelity and temporal coherence. These advances are particularly relevant in the context of autonomous driving, mainly in simulation,

We thank Google Cloud for funding this project by providing access to computational credits for Vertex AI and Gemini.

training, and validation, aiming to produce high-fidelity, temporally coherent, and controllable driving scenarios that capture real-world dynamics, including rare and critical situations. These methods can be categorized into (1) text-to-video [10–12], (2) image-to-video [10, 13, 14], or (3) video-to-video [15, 16]. Recent research in video generation for autonomous driving has emphasized controllable and conditioned generation using both visual and semantic cues. Approaches based on diffusion transformers—such as Cosmos-Transfer1—enable realistic and spatiotemporally coherent videos by conditioning on segmentation maps, depth data, and text prompts relating to driving context (e.g., weather, traffic situations) [14, 17, 18]. Models like DriveDreamer and GEM integrate high-level planning with text or action commands, structuring the generation process for "world models" that underpin robust simulation-to-reality transfer [6, 18, 19]. These methods are increasingly leveraging foundation models and Vision-Language Models (VLMs) to bridge simulation with natural scene understanding, allowing scenario synthesis from abstract or rare prompts (e.g., "child crossing in fog") [17, 20, 21]. However, fine-tuning for specific domains has revealed a trade-off: while visual quality and fidelity often improve, some loss of dynamic semantic accuracy—such as vehicle-pedestrian interactions—can occur if the balance of objectives is not carefully maintained [21].

B. Synthetic datasets for autonomous driving

Synthetic datasets remain crucial for scalable autonomous driving research, allowing exhaustive exploration of rare or hazardous scenarios [2, 17]. Platforms like CARLA [2] and datasets like nuScenes [22] are extended by VLM-based generative pipelines, supporting both low-level (e.g., pixel-level augmentations, weather changes) and high-level (e.g., event-driven, safety-critical) scenario synthesis [6, 17, 18]. Recent advances combine programmatic scene specification (via LLM-generated code) with conditional video generation to efficiently populate training and testing datasets, targeting underrepresented edge conditions [8, 17]. Data augmentation methods range from traditional transforms to LLM-guided scenario recombination and adversarial perturbation [17, 23]. Importantly, these advances have enabled a significant boost in perception and planning model robustness, establishing synthetic datasets as a foundational pillar of modern AV system benchmarking [14, 20].

C. Evaluation of generated visual scenarios

Evaluating generated driving videos for autonomous vehicles demands a multifaceted approach. Classical perceptual metrics like Fréchet Inception Distance (FID) and Fréchet Video Distance (FVD) [24, 25] remain standard for visual realism, but recent studies stress the need for semantic metrics tailored to driving: object-level accuracy (e.g., segmentation/instance tracking), event coverage (e.g., collision, rare maneuver occurrence), and scenario diversity [17, 18, 21]. Human-in-the-loop and AI-judge protocols (e.g., GPT-4o forced-choice or scenario-based QA) are increasingly employed to supplement numerical scores with qualitative validation [21, 26]. Furthermore, "scenario-based evaluation"—requiring downstream perception, planning, and policy modules to operate safely and robustly on generated samples has emerged as a key paradigm [21]. Catastrophic forgetting and reward misspecification during fine-tuning are studied for their effects on both surface realism and behavioral fidelity, underscoring the subtle interplay between visual, semantic, and operational evaluation criteria [21, 27].

III. Method

Since the realism of a textual description provided by a language model is hard to evaluate, we choose to rely on a data augmentation strategy that takes existing real scenarios and modifies specific components to create ECs. This method at least guarantees that a large part of the scene is already grounded in reality and provides a very specific context for the LLM. Our evaluation strategy relies on representing textual descriptions as vector embeddings so that we can calculate a similarity score between two scenarios (Fig. 1). Section III-A focuses on our scenario representation while Section III-B details our generation strategy for driving scenarios. The embeddings, as well as the associated metrics, are described in detail in Section III-C.

A. Structured driving scenarios for explainable scenario generation with an LLM

When directly prompting an LLM to generate loosely defined concepts like ECs, we do not know if the model really understands the output we expect. More generally, a structured formatting of the expected scenarios can help an LLM generate higher-quality and diverse responses. We rely on a 5-layer model (5LM) based on the 6LM proposed by *Scholtes et al.* [9] that decomposes a driving scenario into five different layers, namely (a) road structures, (b) structures surrounding the road, (c) temporary changes to the first two layers, (d) dynamic objects, and finally (e) environmental conditions. This model not only standardizes the representation of a driving scenario, but also allows us to study prompting strategies that target very specific elements of the scenario (Fig. 3). In their model, *Scholtes et al.* use a sixth layer for digital information, such as V2X messages, which are out of scope of the present work. Since our textual synthetic scenarios will be used to generate visual data, we ignore the digital layer which is not visible. Initially, our proposed structure contains only a string of text for each layer, producing an *unstructured 5-layer model*. Based on this model, we define a driving scenario s to be the concatenation of a vector of textual descriptions as:

$$s = \bigoplus_{k=1}^{5} L_k, \qquad (1)$$

where L_k is the textual description of the content of layer k, $\forall k = \{1, 2, \cdots, 5\}$. Eventually, this scenario description is further improved by introducing more details about the content of each layer while constraining the model to fill a template for each layer. This template creates classes of objects

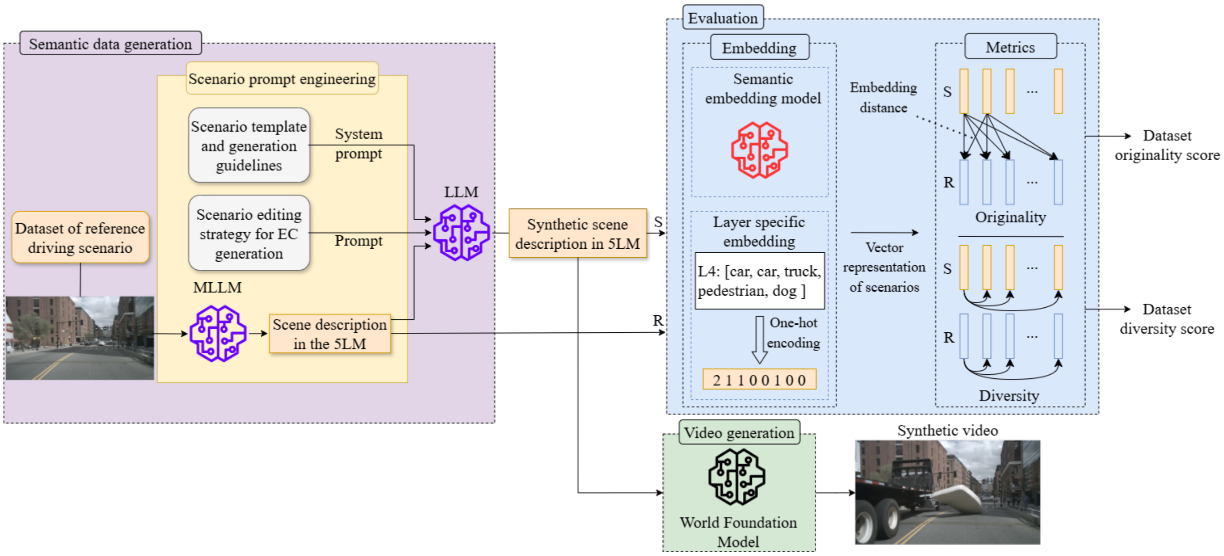

Semantic data generation
Scenario prompt engineering
Scenario template and generation guidelines
System prompt
Scenario editing strategy for EC generation
Prompt
LLM
Dataset of reference driving scenario
MLLM
Scene description in the 5LM
Synthetic scene description in 5LM
S
R
Evaluation
Embedding
Semantic embedding model
Layer specific embedding
L4: [car, car, truck, pedestrian, dog]
One-hot encoding
2 1 1 0 0 1 0 0
Embedding distance
Vector representation of scenarios
Metrics
S
R
Originality
S
R
Diversity
Dataset originality score
Dataset diversity score
Video generation
World Foundation Model
Synthetic video

Fig. 1: Proposed method for generating and evaluating new diverse scenario from real scenes. Our generation strategy relies on editing real scenarios after representing in a 5-layer model.

expected in the layer intended for modifications and indicates the minimum characteristics expected for each object. We refer to this model as a *structured 5-layer model* (Fig. 2).

B. *Scenario generation strategies*

1) Reference scenarios: The method we propose is applicable to textual representations of driving scenarios and not to visual representations such as videos. However, driving datasets consist mostly of image data. Descriptions in common datasets such as NuScenes [22] specify only some of the agents involved without detailed information on the infrastructures or the environment. In order to acquire real reference scenes from which we will generate our scenarios, we use a Multimodal Large Language Model (MLLM) to analyze visual data from a driving dataset and format it according to our model (Fig. 1). For this paper, the nuScenes dataset [22] and the Gemini 2.5 Pro MLLM [28] were used, as they are a popular SotA dataset and model, respectively.

2) Unstructured and non-specific layer editing: Our initial editing strategy consists of prompting the model to edit a specific layer of a real scenario while leaving the other layers unchanged (Fig. 3). We build our prompt by first giving the model a system prompt that explains its role, provides guidelines on how to understand the 5-layers model and explain the concept of EC. At this stage, the scene is decomposed into 5 layers that are just described by unstructured text. Whenever any specific layer L_i is modified, the model is simply asked to modify the given layer:

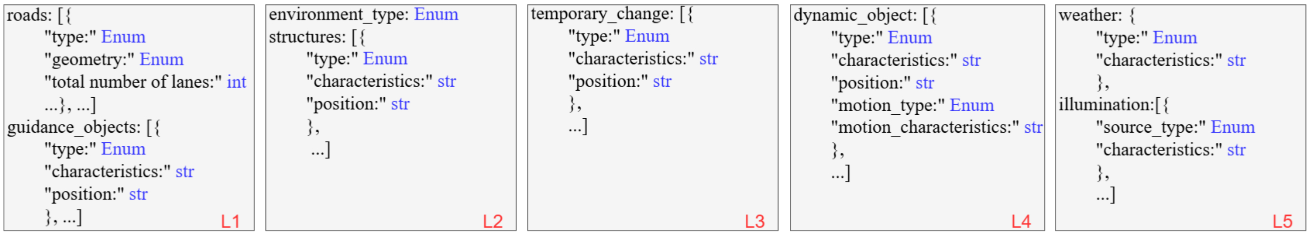
Task: Please only modify the layer specified in the prompt to generate an Edge Case and change nothing in the other layers (MOST IMPORTANT)

3) Structured layer-wise editing: Our second approach consists of providing the model with additional structure within each layer. Using the functionality of the Gemini API, we can force the model output to follow a given JSON template (Fig. 2). Each layer contains specific classes of expected objects and their attributes. Any additional textual details about these objects will be placed in a *'Characteristics'* attribute, which we can use for text-based evaluation, like the previous strategy. This allows us to define layer-wise metrics (see Section III-C) and to introduce a different input prompt well-adapted for each layer. Unlike the previous subsection, when the model is tasked with editing a layer, such as L_4, it will receive an additional task specific to that layer:

Task: Turn this scenario into an Edge Case by modifying only the layer L4 from the input. You should either: - Modify existing dynamic objects, or add new ones with rare and/or challenging characteristics. Look for object that do not belong in such a scenario. - Modify the motion of existing dynamic objects, or add new objects with unique and challenging motion. You may do both if needed, but focus on either the characteristics or the motion of the objects when generating a scenario.

C. *Evaluation metrics*

In order to evaluate the proposed method for generating scenarios, we have to identify relevant evaluation metrics. The final goal of this method would be to generate visual data for EC driving scenarios. However, existing evaluation metrics for synthetic driving data would mostly evaluate the visual performance of the final method used to render these scenarios. We aim to evaluate the relevance of our synthetic scenarios at the textual description level before video generation. Considering that we are looking for ECs, we need to evaluate whether the generated scenario is considered a *rare* and *unexpected* event. Among existing text-based evaluation metrics, *Aasi et al.* proposed two metrics to quantify Diversity and Out-of-Domainness (OOD) [8], based on the calculation of the



```
roads: [{
        "type:" Enum
        "geometry:" Enum
        "total number of lanes:" int
        ...}, ...]
guidance_objects: [{
        "type:" Enum
        "characteristics:" str
        "position:" str
        }, ...]
```

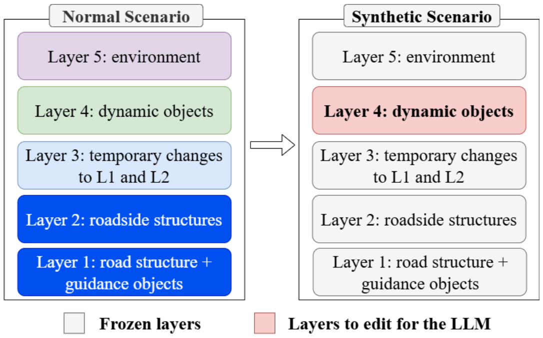
```
environment_type: Enum
structures: [{
        "type:" Enum
        "characteristics:" str
        "position:" str
        },
        ...]
```

```
temporary_change: [{
        "type:" Enum
        "characteristics:" str
        "position:" str
        },
        ...]
```

```
dynamic_object: [{
        "type:" Enum
        "characteristics:" str
        "position:" str
        "motion_type:" Enum
        "motion_characteristics:" str
        },
        ...]
```

```
weather: {
        "type:" Enum
        "characteristics:" str
        },
illumination:[{
        "source_type:" Enum
        "characteristics:" str
        },
        ...]
```

Fig. 2: Snapshot of the structured 5-layer model representation. The Enum fields correspond to a list of available choice for the model, while the str fields are more free section for the model to specify additional information. Some fields like structures (L2), dynamic objects (L4) or illumination (L5) expect a list of all relevant object fitting that category within the scene.



Normal Scenario
Layer 5: environment
Layer 4: dynamic objects
Layer 3: temporary changes to L1 and L2
Layer 2: roadside structures
Layer 1: road structure + guidance objects
Synthetic Scenario
Layer 5: environment
Layer 4: dynamic objects
Layer 3: temporary changes to L1 and L2
Layer 2: roadside structures
Layer 1: road structure + guidance objects
Frozen layers
Layers to edit for the LLM

Fig. 3: Application of the layer model to scenario generation. The LLM is prompted with editing specific layers at a time to increase our control over the generation process. Similarly, the evaluation is also done layer per layer for increased scene understanding

semantic similarity between two given scenarios. Following a similar strategy, we developed two sets of evaluations. The semantic embedding-based metrics use the method from *Aasi et al.*, changing the distance measurement formulas from a sample to a dataset in both of their metrics. It uses the Gemini-embedding-001 model [29] to calculate semantic similarities within the *unstructured 5LM* (Fig. 4). Our second approach relies on the *structured 5LM* to calculate the similarity of the scenarios based on a structure-specific vector representation. Although it is subjective to precisely define what can be considered an EC, these metrics allow us to compare our synthetic data with a real reference, in order to evaluate if they are an EC in regard to that dataset. In both cases, we calculate the similarity between two embeddings using the cosine similarity:

$$\delta(e_i, e_j) = \frac{e_i^T.e_j}{||e_i||\,||e_j||}, \tag{2}$$

where e_i and e_j are the embeddings of two scenarios that are being compared. If e_i and e_j are identical, the cosine value will be 1. The minimum similarity value will be 0 if e_i and e_j are perpendicular, since $e_i^T.e_j = 0$.

Our initial metrics are based on semantic embedding distances between texts. Our goal is to use an embedding model to obtain a quantitative evaluation of how similar our synthetic scenarios are to the reference samples. From that we can quantify how similar our synthetic dataset is to the reference dataset. However, unlike *Aasi et al.* [8] who directly compare the distances between the full description of their scenes and the reference, we rely on the unstructured 5LM to compare each layer of the scenes separately in order to achieve a more precise understanding of the scene's originality. Let $S_{gen} = \{s_{g,1}, \dots, s_{g,M}\}$ and $S_{ref} = \{s_{r,1}, \dots, s_{r,N}\}$ be the set of generated synthetic scenes and real reference scenes respectively. Each scene s is represented in the 5LM as the texts $s_k, \forall k \in [1, 5]$ that represent the descriptions of the corresponding layers. For each layer, we compute the embeddings of both sets $E_{gen,k} = \{e_{g,1,k}, e_{g,2,k} \dots, e_{g,M,k}\}$ and $E_{ref,k} = \{e_{r,1,k}, e_{r,2,k} \dots, e_{r,N,k}\}$, where $e_{g,i,k}$ is the embedding of the k-th layer of the i-th generated scene $s_{g,i} \in S_{gen}$. In this section, the embeddings are computed using Gemini-embedding-001 as it ranks first on several embedding model leaderboards [29] and has been trained on the same data as Gemini-2.5, which we use as the LLM and MLLM of our pipeline.

1) Originality: The Originality score $O^r_{g,j,k}$ of a sample $s_{g,j}$ from a reference set S_{ref} for layer L_k is calculated as:

$$O^r_{g,j,k} = \max_{i \in \{1,\dots,N\}} \delta(e_{r,i,k}, e_{g,j,k}) \tag{3}$$

Unlike us, *Aasi et al.* calculate the OOD score $O^r_{g,j}$ between $s_{g,j}$ and S_{ref} using the sample $e_{r,i}$ with **minimum** similarity instead of **maximum**. Their metrics measure the similarity from the generated sample to the least similar (i.e. the most different) sample within the reference set, while we look at the reference sample that is the closest from our synthetic scenario. We found this to be more informative than the original method, which cannot distinguish if the generated sample is already present in the reference dataset. The final originality scores of the generated layers L_k are then given by the average originality of every new scene with the original set:

$$O^r_{g,k} = \frac{1}{M} \sum_{j=1}^{M} O^r_{g,j,k} \tag{4}$$

2) Diversity: The Originality metric was designed to quantify how different our scenarios are from the reference, since ECs are defined to be unknown events. Similarly, the diversity metric aims to identify whether the scenarios within a set are similar to each other by evaluating the similarity between each

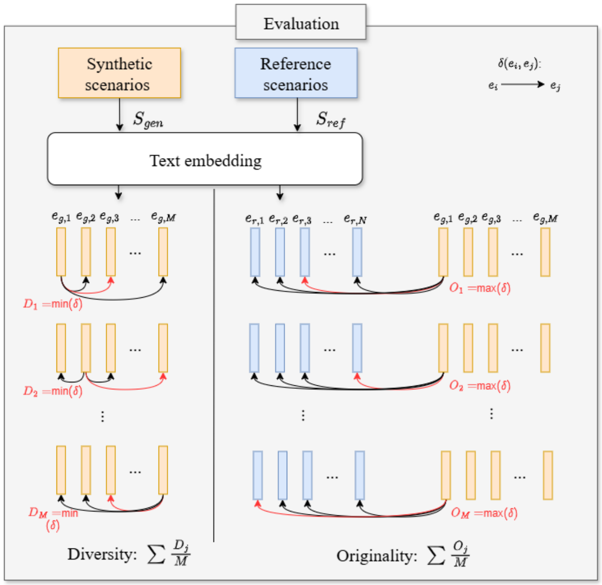


Fig. 4: Diversity and Originality are calculated using the semantic distances between the embeddings of the generated scenarios S_o and the reference scenarios S_b.

sample (Fig. 4). Unlike the previous metric which provides a single score to compare synthetic and reference scene, this metric provide a score for each set separately.

$$D_{g,j,k} = \min_{\substack{j' \in \{1,\dots,M\} \\ j' \neq j}} \delta(e_{g,j',k}, e_{g,j,k}) \tag{5}$$

$$D_{g,k} = \frac{1}{M} \sum_{j=1}^{M} D_{g,j,k} \tag{6}$$

Like the previous metric, we differ from *Aasi et al.* not just with the layer model, but because they calculate diversity as the mean of the **maximum** similarity between samples. However, we found this version to be too penalizing for larger datasets, as increasing the number of samples in a dataset could drastically reduce its diversity, even if it covers a wider range of scenarios. Especially considering that our objective is to generate many new scenarios from a small amount of reference. We will refer to both of these metrics as Diversity(max) and Diversity(min), for the versions of *Aasi et al.* and ours respectively.

3) Layer-wise evaluation: Each layer of the 5LM represents a fundamentally different component of a scenario based on its property, its behavior, or its interaction with the ego vehicle. Although measuring metrics separately for each layer provides some insight into the relevance of the new scenes, our structured model allows us to define a layer-wise evaluation strategy to account for the specificity of each layer. In the structured 5LM (Fig. 2), each component of a layer has a mandatory discrete *type* parameter with predefined and limited possible values. Instead of using the pre-trained embedding model, we rely on this representation to define a structure-specific vector for the layer of a scenario.

For any layer L_k with $k \in \{1, 2, ..., 5\}$ containing a list of objects with a *type* parameter, with C_k possible values. L_k is represented as the vector V_k of dimension C_k. It is defined as the count of objects within L_k in each category. Take a scene s where L_4 = [{'type': 'vehicle', ...}, {'type': 'inanimate object', ...}, {'type': 'vehicle', ...}, {'type': 'pedestrian', ...}] and the available categories are [vehicle, cyclist, pedestrian, animal, inanimate object, other]. $C_k = 6$ and we can embed L_4 as the vector: $V_4(s) = \begin{pmatrix} 2 & 0 & 1 & 0 & 1 & 0 \end{pmatrix}^T$.

This vector can be considered as an embedding of the layer within the type space associated with the expected objects within that layer. Like Section III-C, we can also use that vector to calculate the distance between the distribution of objects present in the layer of each scene. It gives us a new version of diversity and originality which we call Component Diversity (CD) and Component Originality (CO). However, the very small size of these embedding spaces (less than ten dimensions) also reduces the relevance of the diversity metric, as samples are much more likely to reach a similarity score of 0 or 1. Thus, the diversity(min) that we used for previous evaluation would easily stay at 0 in generation experiments, making even 1 outlier too impactful on the Diversity score. Similarly, the diversity(max) of *Aasi et al.* would mostly remain at 1. To find a better metric, we compromised between our approach and *Aasi et al.* by using another diversity score using the average similarity of a sample with every other sample in the same set:

$$D_{g,j,k} = \frac{1}{M-1} \sum_{\substack{j' \in \{1,\dots,M\} \\ j' \neq j}} \delta(V_k(s_{g,j'}), V_k(s_{g,j})) \tag{7}$$

IV. Results and Discussion

A. Experimental setup

Our strategy was implemented on the nuScenes dataset [22]. Specifically, we used the small sample nuScenes-mini containing 10 scenes of twenty seconds. Each video was provided to the MLLM Gemini-2.5-Pro, which was tasked with providing a textual description following our template (see Section III-A). Two types of description were generated, corresponding to the structured and unstructured layer model described in Section III. For each experiment, we used all 10 scenes to generate new scenarios and created 10 new scenarios for each layer, giving us 500 generated descriptions for each experiment. The LLM used is also Gemini-2.5-pro, while the embedding model used for the evaluation metrics is Gemini-embedding-001. The model heat is kept at the default value of 1 for all experiments.

B. Results

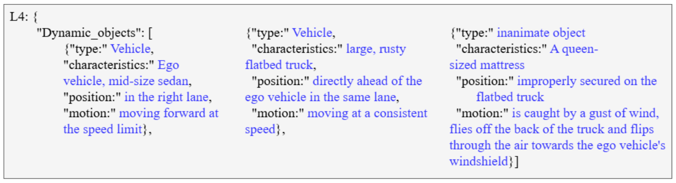
1) Context and Metrics: In Table I, the originality and diversity are calculated for each layer and using both the maximum and minimum semantic similarity values. The OOD-ness with minimum similarity (O(min)) and the diversity with maximum similarity (D(max)) correspond to the metrics presented by *Aasi et al.* [8], while the other two are defined in Equations 4 and 6. All samples in this experiment are generated in the unstructured 5LM. In addition, this experiment evaluates the impact of using the same context to generate scenarios. In one case, the model is called 10 times independently for each layer L_k to generate 1 new scenario by editing L_k. In the other case, the model is directly asked once to provide 10 new scenarios. Overall, it appears that the context's influence is very small, with slightly better results when using the same context. We can also notice that the embedding space of Gemini-embedding-001 puts our textual descriptions close to each other, since the lowest diversity score is only at 0.78. For all of our experiments, the embedding distance between two samples using this model never reached a value lower than 0.7. In addition to the previously mentioned issues of the metrics of *Aasi et al.* (Section. III-C), our metric identify a difference between same and different context more often. The values of D(max) in particular always stays very close to 1 making it hard to use. Thus, we chose to keep our versions of the metrics (i.e. O(max) and D(min)).

TABLE I: For each layer, synthetic scenes were generated either by making 10 independent queries to the model ("$\neq$" context), or all 10 scenes where asked in the same prompt("=" context).

Metric	L1	L2	L3	L4	L5	Total	Context
Aasi: O(min)	0.86	0.84	0.82	0.84	0.80	0.89	$\neq$
	0.86	0.84	**0.81**	0.84	**0.79**	0.89	=
Ours: O(max)	0.91	0.89	0.86	0.90	0.85	**0.97**	$\neq$
	0.91	**0.88**	**0.85**	**0.89**	**0.83**	0.98	=
Ours: D(min)	0.89	0.88	0.78	0.87	0.83	0.90	$\neq$
	0.88	**0.85**	0.78	**0.84**	**0.82**	0.90	=
Aasi: D(max)	0.99	0.99	0.98	0.99	0.99	0.99	$\neq$
	0.99	0.99	0.98	0.99	0.99	**0.97**	=

2) Scenario Structure: In Table II, we compare three different levels of scenario structure to understand the impact of structured scenarios: (a) The unstructured prompt uses a 5-layer model, where the content of each layer is just a string of text. The model is given guidelines on what type of information should go in which layer, but the generation process inside a layer is left free. (b) The 'Soft' structure adds more detail from the previous prompt by specifying a few components the model must mention, as well as some of their characteristics. It adds a specific task prompt depending on which layer is being edited (see Section III-B3). (c) The 'Hard' structure defines a template with classes for each component that specify the information that must be filled. Unlike the previous method, where the soft structured output is still being returned as plain text for each layer, we use Gemini's structured output functionality to force the model to reply with a JSON file following our template (Fig. 2).

Table II shows that all synthetic datasets have a better diversity than the reference scenarios (0.88, 0.81 and 0.88 on synthetic layer 1 against 0.92 on reference). While the soft structure always performs better than the unstructured prompt, the hard structure appears to perform worse than the soft structure, and sometime worse than the unstructured prompt. Although adding more guidance for the model on required information helped its performance, the hard model may suffer from a non-adapted evaluation method. A large part of the text for the hard structure corresponds to the definition of the template, which may impact its diversity, since a large part of its text is identical between samples.



```
L4: {
    "Dynamic_objects": [
        {"type:" Vehicle,
        "characteristics:" Ego
        vehicle, mid-size sedan,
        "position:" in the right lane,
        "motion:" moving forward at
        the speed limit},
{"type:" Vehicle,
 "characteristics:" large, rusty
flatbed truck,
 "position:" directly ahead of the
ego vehicle in the same lane,
 "motion:" moving at a consistent
speed},
{"type:" inanimate object
 "characteristics:" A queen-
sized mattress
 "position:" improperly secured on the
     flatbed truck
 "motion:" is caught by a gust of wind,
flies off the back of the truck and flips
through the air towards the ego vehicle's
windshield}]
```

Fig. 5: Example of an edited layer 4 from scene 1 of nuScenes in the *structured 5LM*. The truck and mattress have been added to the scene.

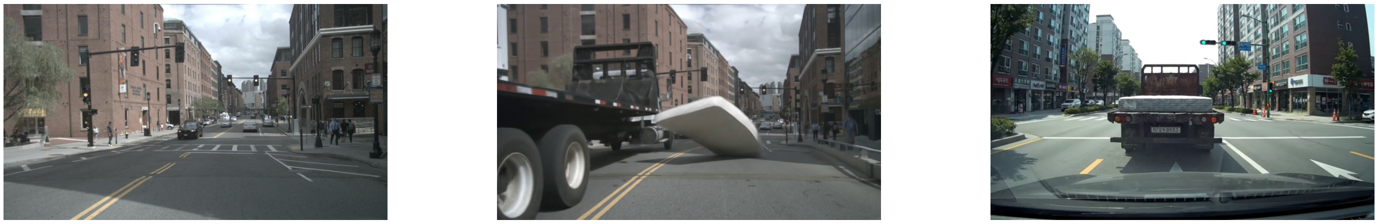
TABLE II: Results of the unstructured 5LM (U), soft (S) and hard (H) structured 5LM. The reference text (R) are the real scenario from nuScenes.

Metric	Structure	L1	L2	L3	L4	L5
O	U	0.91	0.88	0.85	0.89	0.83
	S	0.87	**0.84**	**0.83**	**0.84**	0.82
	H	**0.86**	**0.84**	0.84	0.86	**0.81**
D	R	0.92	0.92	0.89	0.92	0.96
	U	0.88	0.85	0.78	0.84	0.82
	S	**0.81**	**0.78**	**0.76**	**0.78**	**0.80**
	H	0.88	0.83	0.83	0.86	0.81

3) Layer wise Evaluation: The layer-wise evaluation is highly dependent on the structure used for the scenario, and especially the enumeration of the possible types of object, environments, or roads. In the results of Table IV, the generated scenario CD mainly outperforms the reference (0.18 to 0.64 for L2) but occasionally underperforms (0.68 to 0.54 for L1). In addition to the definition of the layer's structure, we found that when generating modified layers, the LLM have a tendency to remove or group objects from the original scene to focus mostly on the new content (see Fig. 5). The average number of dynamic objects per scene in layer 4 falls from 6.4 in the reference to 2.7 in the generated scenes. Either the model does not repeat objects, or group them together as one: 4 car objects can be grouped as: **{'type': 'Vehicle', 'characteristic': 'A mix of sedans and SUVs.', 'position': 'In the opposite lane.'}**. In addition to our layer-specific embedding, we also evaluate the diversity of some components by computing the embedding of the *characteristics* field, which remains pure text, and calculating the Diversity (min) in Table III. For each value, the generated scenes outperform the reference by having a better diversity (i.e., a lower Diversity score) of characteristics among objects, but the gap varies greatly depending on the layer. Note that L3 was underexplored in our study. Since our *structured 5LM* had too few categories for the 'type' field, we chose to leave it as free text which make the CO and CD not applicable to that layer.

TABLE III: The object diversity calculates the diversity score as defined in Section III-C on the corresponding components of that layer using the text in their characteristics fields using Gemini-embedding-001.

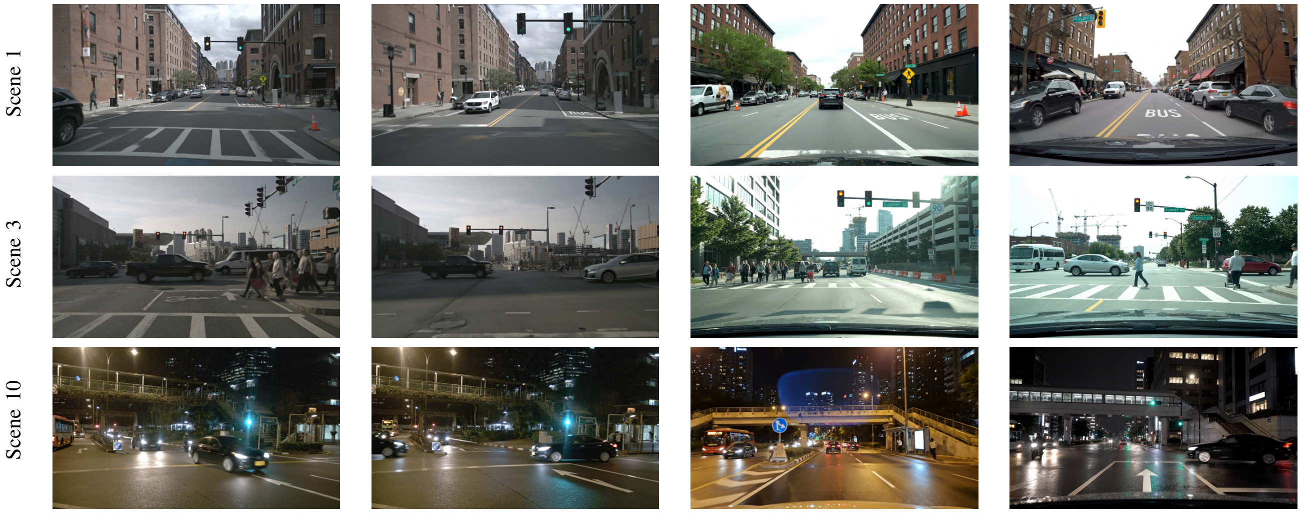
layer	L1		L2		L3	L4		L5	
layer component	roads	guidance	environment	structure	object	object	motion	weather	illumination
Generated scenes	**0.86**	**0.89**	**0.74**	**0.85**	**0.79**	**0.71**	**0.76**	**0.79**	**0.83**
Reference scenes	0.91	0.91	0.83	0.92	0.80	0.74	0.81	0.84	0.87


(a) Front view camera of the scene 1 in the nuScenes dataset, at frame 1.

(b) Video generated using the reference image 6a, with a prompt specifying the edited layer from Fig. 5, at 4 seconds.

(c) Video generated with the same prompt as Fig. 6b without image guidance, at 4 seconds.

Fig. 6: Screenshots of synthetic videos generation using the Veo3-preview WFM on our synthetic scenario shown in Fig.5.



(a) structured image guided

(b) unstructured image guided

(c) structured text guided

(d) unstructured text guided

Fig. 7: Examples of video generation using the Veo3-preview WFM. For 3 different scenes of nuScenes-mini, we generate an 8 seconds video using the structured 5LM description as the prompt (structured) or a description generated without using the 5LM (unstructured). Each image correspond to the same frame, 4 seconds into the video. Those on the left were given the first frame of the original scene as a reference.

TABLE IV: The CD and CO metrics use the vector defined in Section III-C3.

	Metric	L1	L2	L3	L4	L5
Generated Scenes	CO	0.93	0.59	NA	0.85	0.81
	CD	0.68	**0.18**	NA	**0.63**	**0.50**
Reference Scenes	CD	**0.54**	0.64	NA	0.91	0.52

4) Qualitative evaluation of video generation: Using the generated scenario shown in Fig. 5, we use Veo3 [7] to generate video data (Fig. 6). The prompt is our scenario description in the structured 5LM, only adding "*From a dashcam point of view*" at the beginning. Fig. 6 show the generated scenes with (Fig. 6b) and without (Fig. 6c) using the reference image. Although the guided version shows better fidelity to the original, it also struggles with incoherence between the edited scenario and the reference image. Since the truck should be positioned in front of the ego vehicle but is not present on the first frame, it appears on the left of ego vehicle overtaking it. Fig. 7 shows the reconstruction with Veo 3 of three different scenes from nuScenes. When the generation is image-guided, the video generated using the 5LM shows better motion from the ego vehicle and other agents. When the video are not image-guided, while the background is more accurate to the original in the 5LM, there are more hallucinations (the pedestrian coming from the front instead of the side in S3). It is not clear at the moment if this is an issue with the structured 5LM, or with how the WFM should be prompted to respect large prompt.

V. Conclusion

In this paper, we proposed a new synthetic scenario generation strategy focused on semantic descriptions to produce diverse road events. We showed how a structured representation format for driving scenarios can be used to create new editing strategies, but also for more detailed evaluation methods. Circumventing the application-subjective challenge of defining criteria for Edge Case identification, our method focuses on expending the operation domain of any given set of data. The evaluation metrics we propose are based on having a method to calculate the distance between any two scenarios. As such, they can be used with other types of structure as long as that condition is fulfilled. Although our focus is on rare driving scenarios, this allows our proposed method to be very flexible to many specific applications where such structure can be defined. Future works may explore further improvement of our method by exploring new structured models, especially in order to cover the temporal details of the scenes more precisely. Future research should pay attention to studying the behavior of LLM in regards to hallucination or catastrophic forgetting. Finally, the generation of visual data from text description should be improved to better ensure that the most relevant details of the scenes are preserved.

References

[1] A. Elhafsi, R. Sinha, C. Agia, E. Schmerling, I. A. Nesnas, and M. Pavone, "Semantic anomaly detection with large language models," *Autonomous Robots*, vol. 47, no. 8, pp. 1035–1055, 2023. 1

[2] A. Dosovitskiy, G. Ros, F. Codevilla, A. Lopez, and V. Koltun, "Carla: An open urban driving simulator," in *Conference on robot learning*. PMLR, 2017, pp. 1–16. 1, 2

[3] Y. Gao, M. Piccinini, Y. Zhang, D. Wang, K. Moller, R. Brusnicki, B. Zarrouki, A. Gambi, J. F. Totz, K. Storms *et al.*, "Foundation models in autonomous driving: A survey on scenario generation and scenario analysis," *arXiv preprint arXiv:2506.11526*, 2025. 1

[4] C. Chang, S. Wang, J. Zhang, J. Ge, and L. Li, "Llmscenario: Large language model driven scenario generation," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 54, no. 11, pp. 6581–6594, 2024. 1

[5] L. Vater, M. Sonntag, J. Hiller, P. Schaudt, and L. Eckstein, "A systematic approach towards the definition of the terms edge case and corner case for automated driving," in *2023 3rd International Conference on Electrical, Computer, Communications and Mechatronics Engineering (ICECCME)*, 2023, pp. 1–6. 1

[6] X. Wang, Z. Zhu, G. Huang, X. Chen, J. Zhu, and J. Lu, "Drivedreamer: Towards real-world-drive world models for autonomous driving," in *European Conference on Computer Vision*. Springer, 2024, pp. 55–72. 1, 2

[7] Google DeepMind, "Veo 3 model card," https://storage.googleapis.com/deepmind-media/Model-Cards/Veo-3-Model-Card.pdf, 2025. 1, 7

[8] E. Aasi, P. Nguyen, S. Sreeram, G. Rosman, S. Karaman, and D. Rus, "Generating out-of-distribution scenarios using language models," *arXiv preprint arXiv:2411.16554*, 2024. 1, 2, 3, 4, 5

[9] M. Scholtes, L. Westhofen, L. R. Turner, K. Lotto, M. Schuldes, H. Weber, N. Wagener, C. Neurohr, M. H. Bollmann, F. Körtke *et al.*, "6-layer model for a structured description and categorization of urban traffic and environment," *iEEE Access*, vol. 9, pp. 59 131–59 147, 2021. 1, 2

[10] Y. Hu, C. Luo, and Z. Chen, "Make it move: controllable image-to-video generation with text descriptions," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 18 219–18 228. 2

[11] J. Ho, T. Salimans, A. Gritsenko, W. Chan, M. Norouzi, and D. J. Fleet, "Video diffusion models," *Advances in Neural Information Processing Systems*, vol. 35, pp. 8633–8646, 2022. 2

[12] Y. Zhou, D. Zhou, M.-M. Cheng, J. Feng, and Q. Hou, "Storydiffusion: Consistent self-attention for long-range image and video generation," *Advances in Neural Information Processing Systems*, vol. 37, pp. 110 315–110 340, 2024. 2

[13] H. Ni, C. Shi, K. Li, S. X. Huang, and M. R. Min, "Conditional image-to-video generation with latent flow diffusion models," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2023, pp. 18 444–18 455. 2

[14] A. Blattmann, T. Dockhorn, S. Kulal, D. Mendelevitch, M. Kilian, D. Lorenz, Y. Levi, Z. English, V. Voleti, A. Letts *et al.*, "Stable video diffusion: Scaling latent video diffusion models to large datasets," *arXiv preprint arXiv:2311.15127*, 2023. 2

[15] B. Wu, C.-Y. Chuang, X. Wang, Y. Jia, K. Krishnakumar, T. Xiao, F. Liang, L. Yu, and P. Vajda, "Fairy: Fast parallelized instruction-guided video-to-video synthesis," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 8261–8270. 2

[16] F. Liang, B. Wu, J. Wang, L. Yu, K. Li, Y. Zhao, I. Misra, J.-B. Huang, P. Zhang, P. Vajda *et al.*, "Flowvid: Taming imperfect optical flows for consistent video-to-video synthesis," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 8207–8216. 2

[17] Y. Fu, R. Zha, P. Tian, and X. Di, "Llm-based realistic safety-critical driving video generation," *arXiv preprint arXiv:2507.01264*, 2025. 2

[18] S. Gao, J. Yang, L. Chen, K. Chitta, Y. Qiu, A. Geiger, J. Zhang, and H. Li, "Vista: A generalizable driving world model with high fidelity and versatile controllability," *Advances in Neural Information Processing Systems*, vol. 37, pp. 91 560–91 596, 2024. 2

[19] M. Hassan, S. Stapf, A. Rahimi, P. Rezende, Y. Haghighi, D. Brüggemann, I. Katircioglu, L. Zhang, X. Chen, S. Saha *et al.*, "Gem: A generalizable ego-vision multimodal world model for fine-grained ego-motion, object dynamics, and scene composition control," *arXiv preprint arXiv:2412.11198*, 2024. 2

[20] A. Hu, L. Russell, H. Yeo, Z. Murez, G. Fedoseev, A. Kendall, J. Shotton, and G. Corrado, "Gaia-1: a generative world model for autonomous driving (2023)," *arXiv preprint arXiv:2309.17080*. 2

[21] C.-P. Chang, C.-Y. Wang, J. Schmidt, H. Caesar, and A. Pagani, "Seeing clearly, forgetting deeply: Revisiting fine-tuned video generators for driving simulation," *arXiv preprint arXiv:2508.16512*, 2025. 2

[22] H. Caesar, V. Bankiti, A. H. Lang, S. Vora, V. E. Liong, Q. Xu, A. Krishnan, Y. Pan, G. Baldan, and O. Beijbom, "nuscenes: A multimodal dataset for autonomous driving," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 11 621–11 631. 2, 3, 5

[23] L. Russell, A. Hu, L. Bertoni, G. Fedoseev, J. Shotton, E. Arani, and G. Corrado, "Gaia-2: A controllable multi-view generative world model for autonomous driving," *arXiv preprint arXiv:2503.20523*, 2025. 2

[24] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter, "Gans trained by a two time-scale update rule converge to a local nash equilibrium," *Advances in neural information processing systems*, vol. 30, 2017. 2

[25] T. Unterthiner, S. Van Steenkiste, K. Kurach, R. Marinier, M. Michalski, and S. Gelly, "Towards accurate generative models of video: A new metric & challenges," *arXiv preprint arXiv:1812.01717*, 2018. 2

[26] A. Blattmann, R. Rombach, H. Ling, T. Dockhorn, S. W. Kim, S. Fidler, and K. Kreis, "Align your latents: High-resolution video synthesis with latent diffusion models," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2023, pp. 22 563–22 575. 2

[27] G. I. Parisi, R. Kemker, J. L. Part, C. Kanan, and S. Wermter, "Continual lifelong learning with neural networks: A review," *Neural networks*, vol. 113, pp. 54–71, 2019. 2

[28] G. Comanici, E. Bieber, M. Schaekermann, I. Pasupat, N. Sachdeva, I. Dhillon, M. Blistein, O. Ram, D. Zhang, E. Rosen *et al.*, "Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities," *arXiv preprint arXiv:2507.06261*, 2025. 3

[29] J. Lee, F. Chen, S. Dua, D. Cer, M. Shanbhogue, I. Naim, G. H. Ábrego, Z. Li, K. Chen, H. S. Vera *et al.*, "Gemini embedding: Generalizable embeddings from gemini," *arXiv preprint arXiv:2503.07891*, 2025. 4