A Graph-Based Metric on Dataset Diversity and Redundancy Evaluation

Yuankai He
University of Delaware

Hanlin Chen
Oak Ridge National Lab

Ilya Safro
University of Delaware

Weisong Shi
University of Delaware

***Abstract*—Large-scale driving datasets overrepresent common scenes and under-sample rare but safety-critical events, inflating training cost and diluting useful signal. We present S-Score, a lightly-annotated, task-agnostic measure of semantic redundancy. Per-frame scene attributes are automatically extracted by vision–language models (GPT-4o, Qwen2-VL-2B-Instruct) to a standardized schema. We construct a frame–attribute graph and compute five indicators: *attribute overlap, degree imbalance, community persistence, density, and perceived scene risk*. Each indicator is evaluated against an attribute-frequency–matched random baseline that preserves per-attribute frequencies and per-frame attribute counts. Penalties are expressed as relative deviations from this baseline and aggregated with non-negative weights into an unbounded score where larger values indicate lower redundancy. On public driving datasets and standard perception/forecasting tasks, higher S-Score aligns with faster accuracy gains at the same training budget. A controlled counterexample shows that CLIP embeddings with K-center can remove visually similar yet semantically diverse frames. We will release our code, prompts, model weights, and the attribute schema.**

I. Introduction

Autonomous driving systems rely heavily on the quality, diversity, and relevance of training data to support perception, tracking, planning, and control modules [1]–[10]. Despite extensive benchmarks like Argoverse1 [11], KITTI [12], Cityscapes [13], and nuScenes [14], failures such as misclassifying trailer undercarriages or dragging fallen pedestrians highlight that scale alone does not guarantee coverage of rare or safety-critical scenarios. **In fact, large driving datasets tend to *over-represent* common scenes and *under-sample* rare events, inflating training cost and diluting useful signal**.

Existing dataset evaluation methods typically rely on scale-based proxies (e.g., frame count, sensor variety), impact metrics (e.g., citation counts) [1], [2], or visual feature extraction [15]; **however, these methods do not quantify semantic diversity or redundancy**. As a result, data pipelines often suffer from overcollection or costly manual curation.

We address this gap with **S-Score**, a lightly-annotated, task-agnostic measure of semantic redundancy. Per-frame scene attributes are automatically extracted by vision–language models and standardized to a canonical schema without using task labels (no boxes, masks, or tracks). From these attributes, we construct a frame–attribute bipartite graph and compute five indicators: *attribute overlap, degree imbalance, community persistence, density, and perceived scene risk*. Each indicator is *evaluated against an attribute-frequency–matched random baseline* that preserves per-attribute frequencies and per-frame attribute counts. Penalties are expressed as *relative deviations* from this baseline and aggregated with non-negative weights into a score in which larger values indicate lower redundancy and higher diversity.

This abstraction captures both temporal redundancy (e.g., from adjacent frames) and structural redundancy (e.g., similar scenes across locations). We interpret S-Score as an approximation of effective information gain—content diversity minus internal repetition—drawing from Shannon entropy and mutual information.

We evaluate S-Score on public driving datasets — Argoverse1, KITTI, Cityscapes, and nuScenes — using standard tasks spanning 2D detection (YOLOv11, SwinV2), stereo matching (PSMNet), 2D segmentation (DeepLabv3), 3D detection (PointPillars), and motion forecasting (LaneGCN). To isolate redundancy effects, we hold the *training budget* constant by using the same number of training frames and an identical training schedule (epochs/steps, batch size, optimizer, and hardware). Empirically, higher S-Score aligns with faster accuracy gains under these fixed budgets. We also report dataset ranking by using CLIP feature extraction and k center. In addition, a controlled counterexample in which CLIP embeddings with K-center select visually similar yet semantically diverse frames is also included.

Contributions.

- We introduce **S-Score**, a graph-based, lightly-annotated metric for assessing semantic redundancy in high-frame-rate driving datasets.
- We define an **attribute-frequency–matched random baseline** that preserves per-attribute and per-frame degree sequences, and compare indicators using **relative deviations**.
- We aggregate the indicators into an **unbounded score** (higher = less redundancy) and show that higher scores align with faster accuracy gains at **the same number of training frames and under an identical training schedule**.
- We will release **code, prompts, *fine-tuned Qwen2-VL-2B-Instruct model weights***, and the **attribute schema** to facilitate adoption.

This work was supported in part by the National Science Foundation under Grant NSF 2231523.

The remainder of the paper is organized as follows. Section II reviews related work. Section III describes our metric in detail. Section IV presents empirical validation. Finally, Section V concludes the paper with a summary and discussion of future work.

II. Background and Related Works

A. Dataset Evaluation via Proxy Metrics

Large-scale autonomous driving datasets (e.g., Argoverse1, KITTI, Cityscapes, nuScenes) are commonly evaluated using *scale-based proxies*, such as frame counts or sensor configurations [16], [17], or *impact metrics*, such as citation counts. While these indicators reflect quantity or community adoption, they do not capture semantic diversity or internal redundancy. Consequently, practitioners often rely on over-collection or manual curation to ensure that rare or diverse scenarios are represented.

B. Unsupervised Diversity Metrics and Coverage Estimation

Unsupervised methods have been developed to assess dataset diversity without requiring supervision. These include: (1) *random-sampling baselines*, which assume utility scales with dataset size but ignore content distribution; (2) *feature-space coverage* metrics such as k-means clustering or pairwise distances over pretrained CNN embeddings [15], which can reflect low-level visual variety but fail to capture semantic novelty; and (3) *core-set selection* via farthest-point sampling [15], which emphasizes geometric diversity in feature space but is computationally intensive. Recently, entropy-based filtering [18] has been used to identify redundancy based on feature distributions. However, these techniques operate on instance-level embeddings and do not account for high-level scene semantics or structural repetition that are especially critical for high-frame-rate driving videos.

C. Task-Coupled Data Valuation and Active Selection

A large body of work focuses on model-driven or task-coupled dataset selection. Neural pruning methods discard low-contribution samples based on training gradients or loss dynamics [19], while active learning identifies uncertain or diverse samples for labeling [20]–[22]. Generative models have also been used to augment or reweight rare scenarios in accelerated evaluation frameworks [23], [24]. More recent techniques estimate sample-level data value using a combination of model complexity and mutual information [25], or apply difficulty-based metrics for curriculum learning [26]. While effective in specific training pipelines, these approaches typically require supervision, model access, or task-specific feedback, and are therefore not applicable to pre-training dataset diagnostics.

D. Information-Theoretic and Graph-Based Perspectives

Shannon entropy and mutual information offer principled tools for quantifying diversity and redundancy in data streams [27]. Such concepts have informed frameworks for sensor placement and submodular selection [28], which aim to optimize information gain under resource constraints. However, most existing methods depend on explicit probabilistic models or ground-truth labels, and they do not incorporate structured representations such as graphs or co-occurrence of semantic attributes. Our work draws inspiration from these principles and extends them to a graph-theoretic formulation that captures structured relationships between high-level scene elements.

III. Methodology

We compute **S-Score**, a minimally supervised, graph-based metric designed to quantify semantic diversity and structural redundancy in high-frame-rate driving datasets. S-Score proceeds in five stages, shown in Fig 1, which collectively implement three logical phases: attribute extraction, graph construction, and redundancy scoring.

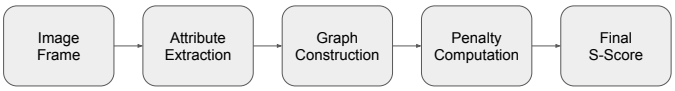

Fig. 1. High-level pipeline.

A. Attribute Summarization

Before any structured analysis can begin, we need to standardize the terminology and criteria. Using 1,000 human-selected (based on coverage and diversity) frames from the Waymo dataset, a VLM is instructed to output scene-level attributes. The integer severity score is calculated based on the proximity of objects to the ego-vehicle and based on the potential danger (i.e. likelihood of a passenger opening the doors of a parked vehicle and how dangerous it is). Then, combined with expert domain knowledge, the scene-level attributes are consolidated into a knowledge base consisting of 28-key category/sub-category with 350+ attributes. The 28-key category/sub-category covers environment, road geometry, background vehicle behaviors, ego vehicle behaviors, vulnerable road user behaviors, sensor states, and a severity score.

B. Scene Attribute

For the dataset under evaluation (DUE), each image frame I_f is processed using a VLM via structured prompt engineering, producing a 28-key JSON annotation covering scene-level attributes. The VLM is instructed to only choose attributes from the knowledge base. The VLM is also instructed to notify the human annotator if it detects a new attribute (e.g. no dogs present in the Waymo dataset but present in Cityscapes). Manual correction is required to correct any incorrect attributes and to update the knowledge base.

To assess the reliability of the VLM's scene attribute extraction, we compared its outputs against human annotations on 500 randomly sampled frames under varied conditions (e.g., day/night, urban/highway, weather). As shown in Table I, GPT-4o achieved over 90% agreement, with most errors arising from ambiguous maneuvers or occlusions. Out of $\sim 10,000$ predictions, only $\sim 1\%$ required correction by a human

TABLE I
VLM ANALYSIS F1-SCORE AND INFERENCE TIME (S). THE GROUND TRUTH IS PROVIDED BY AN HUMAN ANNOTATOR.

	Scene	Time	Weather	Road	Lane	Signs	Vehicles	Peds	Dir.	Ego	Vis.	Camera	Severity	Processing Time
ChatGPT-4o	0.98	0.99	0.95	0.99	0.95	0.95	0.99	0.98	0.93	0.93	0.99	0.98	0.99	45.63s
Qwen2-VL-2B-INS	0.71	0.99	0.74	0.99	0.75	0.76	0.67	0.81	0.88	0.75	0.96	0.99	0.40	44.58s

annotator. Notably, GPT-4o struggled with ego-vehicle motion, likely due to the use of static, shuffled images. It outperformed GPT-4 slightly in recognizing other vehicles' actions, though both models had difficulty with logical dependencies (e.g., associating rain with wet roads) unless explicitly prompted. Qwen2-VL-2B-Instruct is a much smaller VLM model and requires a lot more manual corrections. It fails miserably on predicting the `severity` due it is lack of chain-of-thought capabilities.

Since all outputs were verified and corrected if needed, the final attribute matrix is effectively noise-free, and we do not perform additional sensitivity analysis on VLM errors.

C. Unweighted Attribute Co-Occurrence Graph

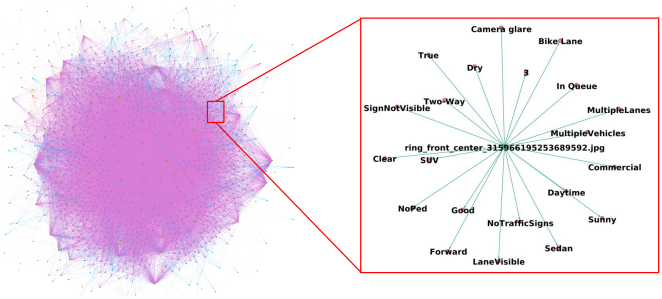

Fig. 2. Graph-based dataset modeling of Argoverse1. Zooming in, we see how each image node is connected to multiple feature nodes.

We encode the dataset as a directed, unweighted graph $G = (V, E)$ where semantic relationships among scenes, images, and attributes are represented structurally. In other words, the dataset graph G and is a root node with directed edges to different G_{img} as shown in Fig 2. The node set comprises:

$$V = V_s \cup V_{img} \cup V_c \cup V_{sc} \cup V_a$$

where V_s denotes scenario clusters, V_{img} image frames, V_c attribute categories (e.g., weather), V_{sc} subcategories (e.g., traffic_sign_visibility), and V_a atomic attributes (e.g., moving). Edges capture co-occurrence or hierarchy: $(v_i \rightarrow a) \in E$ iff attribute $a \in \mathcal{A}_f$ for image v_i, and $(c \rightarrow sc) \in E$ links categories to their subtypes. For example, a frame annotated with "rain," "night," and "pedestrian" connects to those attribute nodes. This structure captures both temporal redundancy (across adjacent frames) and structural redundancy (across scenes), forming the basis for dataset-level analysis.

D. Graph-Based Redundancy Metrics

To quantify redundancy, we compute five graph-derived metrics $M_i(G)$, each capturing a distinct signal:

Jaccard Similarity. For image pair $G_{img,A}, G_{img,B}$), compute:

$$J(G_{img,A}, G_{img,B}) = \frac{|G_{img,A} \cap G_{img,B}|}{|G_{img,A} \cup G_{img,B}|}$$

A high average Jaccard index across all frames suggests low content variability and repeated attribute combinations.

Degree Centrality [29] reflects attribute frequency. For node v where $v \in V_a$, we compute:

$$C_{\text{in}}(v) = \frac{\text{InDeg}(v)}{N-1}, \quad C_{\text{out}}(v) = \frac{\text{OutDeg}(v)}{N-1}$$

Attributes with high in-degree are frequent across the dataset, indicating possible redundancy.

Modularity [30] measures how well attributes cluster into distinct communities:

$$Q = \frac{1}{m} \sum_{i,j} \left[A_{ij} - \frac{k_i^{\text{out}} k_j^{\text{in}}}{m} \right] \delta(c_i, c_j)$$

where A_{ij} is the adjacency matrix, m is the number of edges, and δ is the Kronecker delta on community assignment. High modularity implies well-separated scene clusters and higher diversity

Graph Density. Defined as:

$$D = \frac{E}{N(N-1)}$$

Lower density implies more unique co-occurrence patterns and lower redundancy.

Risk Imbalance. We compute the normalized entropy over the empirical distribution of risk levels $\{r_f\}$. The risk level is the `severity` score bucketed into low/medium/high with the thresholds at 1–3/4–7/8–10:

$$H(r) = - \sum_{r \in \{\text{low,med,high}\}} p_r \log p_r$$

Lower entropy indicates over-representation of a single risk level, suggesting biased or repetitive scenario coverage.

Our graph-based formulation approximates semantic diversity using principles from information theory. Degree centrality and attribute co-occurrence reflect frequency and support size, which relate to entropy, while risk imbalance directly measures entropy over risk levels. This design is inspired by submodular information functions used in data valuation [27], [28], but we adopt a structural—rather than probabilistic—approach to avoid relying on task labels or generative models.

Even though frames are processed in random order, semantic repetition (e.g., recurring intersections or highway scenes)

results in overlapping attribute sets. Metrics like Jaccard similarity implicitly capture this temporal redundancy, allowing S-Score to reflect both structural and temporal repetition without explicit sequence modeling.

E. Marginal-Preserving Null Reference

To compare graphs on a common footing, we use a *marginal-preserving null* that randomizes attribute co-occurrences while preserving per-category marginals.

Sampling. Partition $\mathcal{A}$ into (i) *singleton* categories (exactly one attribute per frame; e.g., `time_of_day`, `weather`) and (ii) *multi-select* categories (zero or more; e.g., `vehicle_types`). For each synthetic frame $\tilde{v}$: (1) for each singleton category c, sample one attribute from the empirical distribution of c; (2) for each multi-select category c, sample a count n_c from the empirical distribution of counts for c, then sample n_c distinct attributes without replacement according to their empirical frequencies. Repeat to match $|V_{img}|$ and build a synthetic graph $\tilde{G}$. Compute the same metrics on $\tilde{G}$ to obtain reference values M_i^{ref} (optionally averaged over K i.i.d. null graphs). (3) The null graph is checked by the human annotator to ensure no unrealistic combination is included.

Unbounded penalties. For each metric $M_i(G)$, define the relative (unbounded) penalty

$$P_i \;=\; \frac{M_i(G) - M_i^{\text{ref}}}{M_i^{\text{ref}} + \varepsilon},$$

with a small ε for numerical safety. Positive P_i means "more redundant than the null" (e.g., higher Jaccard/modularity/density/degree concentration); negative P_i means "less redundant than the null" (e.g., higher risk entropy).

Composite score. Aggregate as

$$S \;=\; 1 - \sum_{i \in \{\text{sim,deg,mod,dens,risk}\}} w_i\, P_i, \qquad \sum_i w_i = 1,$$

yielding an *unbounded* S; larger S indicates less redundancy / greater semantic diversity relative to the null.

Weight Selection. The five graph-based penalties capture distinct aspects of redundancy (e.g., attribute overlap, risk imbalance), and their relative importance may vary depending on the downstream application. For example, risk distribution may be more critical in safety-sensitive tasks like planning, while structural diversity may dominate in perception or mapping contexts. To support such flexibility, S-Score is defined as a weighted sum $S = 1 - \sum_i w_i P_i$, where users may adjust the weights w_i to reflect task-specific priorities.

Time series datasets. While nuScenes and Cityscapes already randomizes the ordering of the images, Argoverse1 and KITTI provides ordered frames for each scenario divided into different folders. This also allows us to compute not only the scene-level diversities but also the scenario-level diversities, allowing us to capture temporal dependencies and causal relationships essential for understanding driving sequences.

F. Empirical Validation via Subsampling

To evaluate whether S reflects real dataset utility, we perform uniform subsampling on each dataset at 10% intervals (10–100%) and train five representative models across five core tasks: YOLOv11 and SwinV2 for 2D object detection, PSMNet for stereo matching, DeepLabv3 for semantic segmentation, Pointpillars for 3D object detection, and LaneGCN for motion forecasting.

For each dataset-model pair, we record the performance curve $m(f)$ as a function of subsample fraction $f \in \{10\%, \ldots, 100\%\}$. We then compute the data utility score as the percentage of metric (mAP, mIoU, F1, aAcc) improvements. A dataset with a higher S-Score should observe a higher overall utility score than that of a lower scoring dataset.

$$Utility = \frac{(m(f) - m(10))/m(10)}{(f - 10)/f}$$

G. Limitation

A more detailed VLM and human annotation comparison is available but due to page limits we are unable to include it in this paper. A weights stability test is out of the scope of this paper as the weights are tunable and will result in different rankings. A noise-injection stress test to flip a random p% of attribute bits to test S-Score's ability to withstand noisy attributes is out of the scope of this paper and therefore is not performed. Scalability and practicality concerns are only partially addressed with the locally trained open-source Qwen2-VL-2B-Instruct model due to limited GPU resources; however, a sanity check on other proprietary models such as Claude 4.0 Sonnet, Gemini-2.5 Pro 06-05, Sonar, and Grok-3 and open-source models such as Qwen3.5-32B shows similar results with ChatGPT-4o for attribute extraction. It is safe to assume that the S-Score is able to scale well to very large datasets without incurring too much manual curation efforts.

IV. Experiment Results

We validate the utility of our proposed metric S by measuring its correlation with model-training efficiency on five core autonomous driving tasks on KITTI, Cityscapes, nuScenes, and Argoverse1. *Utility* is defined as the rate of model performance improvement under increasing dataset size. Higher S-Score aligns with faster accuracy gains at the same training budget. A controlled counterexample shows that CLIP embeddings with K-center can select visually diverse yet semantically repetitive frames.

- **2D object detection** with YOLOv11n [31] on all four datasets.
- **2D object detection** with vision transformer SwinV2-tiny [32], [33] on Argoverse1, KITTI, and Cityscapes.
- **2D object segmentation** with DeepLabv3 [34], [35] on KITTI and Cityscapes.
- **3D object detection** with PointPillars [36], [37] on KITTI and nuScenes.
- **Stereo matching** with PSMNet [38] on Argoverse1, KITTI, and Cityscapes.
- **Motion forecasting** with LaneGCN [39] on Argoverse1.

TABLE II
DATASET RANKING BY S-SCORE.

Name	S-Score	Frames	Categories
Argoverse1	0.2982 (4)	600K	15
Cityscapes	0.3752 (3)	12K	30
KITTI	0.3767 (2)	41K	8
nuScenes	0.4617 (1)	40K	23

TABLE III
DATASET RANKING BY DOWNSTREAM TASKS. RANKING IS BASED ON UTILITY GAINS. A HIGHER RANKING INDICATES A LARGER UTILITY GAIN.

Name	YOLO	SwinV2	Deeplab	PointPillars	PSMNet
Argoverse1	4	3	N/A	N/A	1
Cityscapes	3	2	2	2	2
KITTI	2	1	1	1	3
nuScenes	1	N/A	N/A	N/A	N/A

A. Composite S-Score

The weights chosen to calculate the S-Score are $w1 = 0.3, w2 = 0.2, w3 = 0.2, w4 = 0.1, w5 = 0.2$ These weights are chosen to more heavily punish structural similarity, being lenient on graph density due to down-sampling, and to punish evenly for temporal similarity, degree imbalance, and perceived risk distribution.

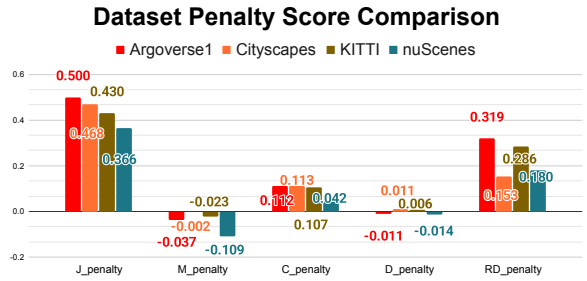

Fig. 3. Dataset penalty score comparison for Argoverse1, Cityscapes, KITTI, and nuScenes. Higher penalty values indicate greater redundancy or imbalance, as measured by Jaccard similarity ($J_{penalty}$), modularity ($M_{penalty}$), centrality ($C_{penalty}$), density ($D_{penalty}$), and risk distribution imbalance ($RD_{penalty}$).

Figure 3 presents the penalty scores for each dataset. Table II shows the datasets' S-Score, ranking, and characteristics. The results reveal key differences in dataset structure:

- **Argoverse1 has the highest redundancy** (high $J_{penalty}$ and $RD_{penalty}$), indicating excessive repetition in driving scenarios.
- **nuScenes has the highest modularity score** ($M_{penalty}$ is lowest), meaning its driving scenarios are more distinct and diverse.
- **Cityscapes and KITTI show moderate redundancy**, but their **graph density values** ($D_{penalty}$) suggest better-balanced feature distributions compared to Argoverse1.

Table III shows the datasets ranked by training efficiencies from different downstream tasks. The training efficiency of the downstream models correlates strongly with the dataset ranking by S-Score. A dataset with a higher S-Score demonstrates a higher training efficiency than that with a lower S-Score. PSMNet benefits more from repetitive data, hence dataset with a lower S-Score has a higher training efficiency.

TABLE IV
EXPERIMENT SETUP AND TRAINING CONFIGURATION.

Experiment setup: 4× NVIDIA RTX 2080 Ti (11 GB each)		
Task	**Model**	**Parameters**
2D Object Detection	YOLOv11n	Default parameters
2D Object Detection	SwinV2	Default parameters
Semantic Segmentation	DeepLabv3	Default parameters
Stereo Matching	PSMNet	Default parameters
3D Object Detection	PointPillars	Default parameters
Motion Forecasting	LaneGCN	Default parameters

B. Experiment Setup

Table IV lists the equipment and hyperparameters used during training.

C. 2D object detection with YOLOv11n

TABLE V
A SUMMARY OF YOLOV11N'S PERFORMANCE ACROSS DATASETS AND SIZES. PERFORMANCE IS MEASURED BY OVERALL MAP50-90 AND F1-SCORES.

Dataset	**Size**	**mAP50-90**	**F1**	**Utility**
Argoverse1	39,515	**0.188**	**0.381**	0.001
	967	0.182	0.379	N/A
Cityscapes	2,976	**0.271**	**0.545**	0.072
	1,488	0.250	0.485	0.131
	297	0.164	0.346	N/A
KITTI	5,985	**0.625**	**0.813**	0.084
	2,992	0.564	0.770	0.146
	598	0.356	0.587	N/A
nuScenes	11,178	**0.471**	**0.710**	0.114
	5,589	0.365	0.615	0.165
	1,117	0.220	0.460	N/A

Table V demonstrates how the YOLO model's performance gains decreases as it is trained on more data from each dataset. The drastic decrease in model performance gain when training YOLOv11n on the Argoverse1 correlates with the lowest S-Score and largest penalty calculated shown in Fig 3. The S-Score ranking for Cityscapes, KITTI, and nuScenes also correlates with the rate of which the YOLOv11n model performance gains as more data is added to the training set.

D. 2D object detection with vision transformer SwinV2-tiny

Table VI shows that Argoverse1, Cityscapes, and KITTI follow the utility trends predicted by their S-Scores.

E. 2D Segmentation with DeepLabv3

Table VII shows that KITTI demonstrates greater utility gains as more data is added. This suggests that KITTI's smaller but more varied annotation set offers steeper performance improvement per sample. This aligns with the S-Score ranking, where KITTI scored slightly higher than Cityscapes. These results highlight that S-Score captures not just diversity but

TABLE VI
A SUMMARY OF SWINV2-TINY'S PERFORMANCE ACROSS DATASETS AND SIZES. PERFORMANCE IS MEASURED BY THE OVERALL MAP50-90 RESULTS.

Dataset	Size	mAP50-90	Utility
Argoverse1	39,515	**0.396**	0.016
	19,758	0.360	0.011
	3,952	0.345	N/A
Cityscapes	2,976	**0.305**	0.046
	1,488	0.290	0.085
	297	0.216	N/A
KITTI	5,985	**0.571**	0.059
	2,992	0.548	0.117
	598	0.373	N/A

TABLE VII
A SUMMARY OF DEEPLABV3'S PERFORMANCE ACROSS DATASETS AND SIZES. PERFORMANCE IS MEASURED BY THE IOU AND AACC(OVERALL ACCURACY).

Dataset	Size	mIoU	aAcc	Utility
Cityscapes	2,976	78.44	96.08	0.072
	2,678	**78.92**	**96.12**	0.078
	1,488	77.99	96.02	0.131
	297	78.75	96.07	N/A
KITTI	180	**31.19**	82.24	0.084
	162	31.18	**82.56**	0.085
	90	29.25	80.03	0.146
	18	19.82	70.85	N/A

also the informativeness of additional data for segmentation tasks.

F. 3D Object Detection with PointPillars

TABLE VIII
A SUMMARY OF POINTPILLARS' PERFORMANCE ACROSS DATASETS AND SIZES. PERFORMANCE IS MEASURED BY THE OVERALL 3D-AP40(HARD) AND BEV-AP40(MODERATE).

Dataset	Size	3D-40H	BEV-40M	Utility
KITTI	3,712	**57.43**	68.32	0.004
	3,340	57.14	**68.74**	0.004
	1,856	56.61	67.95	0.006
	371	55.22	68.14	N/A
Dataset	**Size**	**mAP**	**NDS**	**Utility**
nuScenes	28,130	**0.320**	**0.391**	0.201
	25,317	0.317	0.390	0.223
	14,065	0.149	0.295	0.077
	2,813	0.114	0.246	N/A

Table VIII shows that nuScenes yields significantly higher utility scores than KITTI for 3D object detection with PointPillars. This aligns with their S-Score rankings—nuScenes having the highest and KITTI a mid-range score, supporting the claim that S-Score reflects semantic diversity and dataset efficiency. The limited utility gains from KITTI further validate that lower S-Scores correlate with redundant or less informative data.

TABLE IX
A SUMMARY OF PSMNET'S PERFORMANCE ACROSS DATASETS AND SIZES. PERFORMANCE IS MEASURED BY THE MODEL'S 3-PIXEL ERROR PERCENTAGE.

Dataset	Size	3px Error	Utility
Argoverse1	8,142	**3.845**	0.052
	7,298	4.717	0.043
	4,071	6.338	0.029
	884	7.173	N/A
Cityscapes	2,976	**1.203**	0.028
	2,678	1.249	0.028
	1,488	1.288	0.051
	297	1.618	N/A
KITTI	160	**2.154**	0.027
	144	2.525	0.028
	80	2.676	0.044
	16	3.254	N/A

G. Stereo matching with PSMNet

Table IX shows fine-tuning PSMNet on three real-world datasets. While Argoverse1 achieves the largest performance gains for PSMNet as more data is added, this result does not contradict its low S-Score. Rather, it reflects the nature of what S-Score captures. A low S-Score indicates high structural and semantic repetition within the dataset. In the case of PSMNet, high repetition is beneficial to performance gains. PSMNet learns to estimate disparity through patterns in spatial structure, and repeated exposure to similar geometric layouts helps the model converge more quickly and reliably [40]. In this sense, the improved performance is a sign that the model is effectively exploiting redundant structure. The fact that S-Score correctly identifies Argoverse1 as highly repetitive, and that this repetition aligns with improved performance in a task that benefits from it, further supports the interpretability of the metric.

H. Motion forecast with LaneGCN

TABLE X
A SUMMARY OF LANEGCN'S PERFORMANCE ON ARGOVERSE1 WITH DIFFERENT DATA SIZES. PERFORMANCE IS MEASURED BY ADE (AVERAGE DISPLACEMENT ERROR) AND FDE (FINAL DISPLACEMENT ERROR).

Dataset	Size	ade	fde	Utility
Argoverse1	205,942	**0.676**	**1.01**	0.02
	185,347	0.723	1.112	0.018
	102,971	0.726	1.118	0.017
	20,594	0.844	1.383	N/A

As shown in Table X, reducing the dataset from 205k to 185k frames causes only a modest degradation, while dropping to 20k frames significantly harms performance. This supports the need for diversity-aware selection.

I. Validation of Random Graph

Random synthetic graphs preserve marginal attribute frequencies but break higher-order co-occurrence, validating their

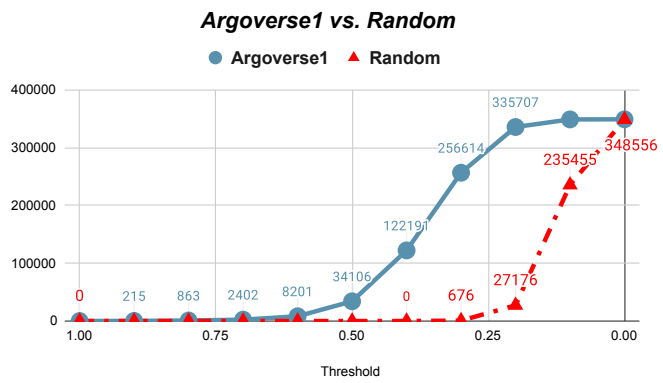

Fig. 4. Jaccard similarity CDF comparison: Argoverse1 vs. Random. 1 indicates the pair of images contains the exact same attributes. 0 indicates that the pair of images share no attributes.

TABLE XI
DATASET RANKING BY CLIP FEATURE EXTRACTION AND K CENTER USING COVERAGE RADIUS (CR), AREA UNDER COVERAGE CURVE (AUC), AND PAIRWISE MEAN DISTANCE (PW)

Name	CR	AUC	PW
Argoverse1	0.1245 (4)	13.2670 (3)	0.1979 (3)
Cityscapes	0.1362 (2)	23.5151 (2)	0.1906 (4)
KITTI	0.1333 (3)	13.1077 (4)	0.2023 (2)
nuScenes	0.1426 (1)	38.8818 (1)	0.2113 (1)

use as null models. Fig 4 shows the CDF of structurally similar images when comparing Argoverse1 versus its respective random synthetic dataset.

J. S-Score sensitivity

To evaluate the sensitivity of the composite S-Score, we perform a sanity test on the downsampled Argoverse1 dataset (836 frames). Removing six frames whose attribute combinations were unique (0.72% of Argoverse1) decreased S from 0.5585 to 0.5565 ($\Delta = -0.0020$), indicating responsiveness to loss of rare semantics.

K. CLIP Feature Extraction and K Center

CLIP focuses on visual similarity and can mistakenly miss out on valuable data, as shown in Fig 5. While Fig 5 (a),(b),(c) may visually look similar, they contain drastically different semantic information such as lane markings and different moving vehicles. This justifies that CLIP feature extraction is not a good tool to evaluate a dataset's diversity and redundancy or select data from one.

As shown in Table XI, we observe rank inversions across evaluation metrics: although CLIP-based selection ranks Argoverse1 as most redundant, AUC identifies KITTI as worst, and PM flags Cityscapes. Rank correlations between CLIP and AUC/PM are low. These discrepancies indicate that generic CLIP embeddings are insufficient to characterize redundancy/diversity in autonomous-driving data. In contrast, S-Score—built from driving-relevant attributes and a marginal-preserving null—better tracks downstream utility.

(a)

(b)

(c)

Fig. 5. CLIP focuses on the overall visual appearance. It focuses on the color, lighting, layout, texture, and object configuration. From a time series of 34 images, CLIP selected only the top image (a). While filtering image with a Jaccard threshold at 0.5 yields the bottom images (b) and (c). Even though (a) and (b) have a Jaccard Similarity of 0.852, (a) and (c) have a Jaccard Similarity of only 0.417. (b) and (c) have a Jaccard Similarity of 0.361.

V. CONCLUSION AND FUTURE WORKS

We introduced **S-Score**, a minimally supervised, graph-theoretic metric that quantifies semantic diversity and redundancy in high-frame-rate autonomous driving datasets. It uses a vision-language LLM to extract scene attributes, builds a directed co-occurrence graph, and combines five normalized penalties (Jaccard similarity, degree centrality, modularity, density, and risk imbalance) against a randomized dataset-specific baseline into one interpretable score.

S-Score is computed at the dataset level, requiring neither task labels nor model training, making it well-suited for early dataset triage and pretraining diagnostics. It supports practical curation decisions (removing redundancy, targeting missing coverage) and correlates with downstream performance across four AV tasks; we will further report how ΔS-Score tracks Δperformance (accuracy/ADE/FDE) as data is incrementally added under a fixed training schedule.

All code, prompts, and analysis scripts are available at https://github.com/Croquembouche/UDrive . While we currently use GPT-4o for attribute extraction, preliminary tests with Claude 4.0, Gemini-2.5, Grok-3, and Qwen3.5-32B indicate similar results are achievable with open-source VLMs.

Limitations and Future Work. We validate S-Score on four datasets and five perception tasks; extending it to multimodal fusion and planning is future work. We also plan sensitivity analysis of graph-metric weights and a systematic eval-

uation of VLM annotation reliability, including stress/noise tests, to characterize robustness and failure modes.

REFERENCES

[1] C. Gao, G. Wang, W. Shi, Z. Wang, and Y. Chen, "Autonomous driving security: State of the art and challenges," *IEEE Internet of Things Journal*, vol. 9, no. 10, pp. 7572–7595, 2021.

[2] L. Chen, P. Wu, K. Chitta, B. Jaeger, A. Geiger, and H. Li, "End-to-end autonomous driving: Challenges and frontiers," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024.

[3] K. Muhammad, A. Ullah, J. Lloret, J. D. Ser, and V. H. C. de Albuquerque, "Deep learning for safe autonomous driving: Current challenges and future directions," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 7, pp. 4316–4336, 2020.

[4] I. Yaqoob, L. U. Khan, S. A. Kazmi, M. Imran, N. Guizani, and C. S. Hong, "Autonomous driving cars in smart cities: Recent advances, requirements, and challenges," *IEEE Network*, vol. 34, no. 1, pp. 174–181, 2019.

[5] Y. Xiao, F. Codevilla, A. Gurram, O. Urfalioglu, and A. M. López, "Multimodal end-to-end autonomous driving," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 1, pp. 537–547, 2020.

[6] M. Uricár, D. Hurych, P. Krizek, and S. Yogamani, "Challenges in designing datasets and validation for autonomous driving," *arXiv preprint arXiv:1901.09270*, 2019.

[7] L. Li, W. Shao, W. Dong, Y. Tian, Q. Zhang, K. Yang, and W. Zhang, "Data-centric evolution in autonomous driving: A comprehensive survey of big data system, data mining, and closed-loop technologies," *arXiv preprint arXiv:2401.12888*, 2024.

[8] D. Feng, C. Haase-Schütz, L. Rosenbaum, H. Hertlein, C. Glaeser, F. Timm, W. Wiesbeck, and K. Dietmayer, "Deep multi-modal object detection and semantic segmentation for autonomous driving: Datasets, methods, and challenges," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 3, pp. 1341–1360, 2020.

[9] Y. Kang, H. Yin, and C. Berger, "Test your self-driving algorithm: An overview of publicly available driving datasets and virtual testing environments," *IEEE Transactions on Intelligent Vehicles*, vol. 4, no. 2, pp. 171–185, 2019.

[10] J. Janai, F. Güney, A. Behl, and A. Geiger, "Computer vision for autonomous vehicles: Problems, datasets and state of the art," *Foundations and Trends® in Computer Graphics and Vision*, vol. 12, no. 1–3, pp. 1–308, 2020.

[11] M.-F. Chang, J. Lambert, P. Sangkloy, J. Singh, S. Bak, A. Hartnett, D. Wang, P. Carr, S. Lucey, and D. Ramanan, "Argoverse: 3d tracking and forecasting with rich maps," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 8748–8757.

[12] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, "Vision meets robotics: The kitti dataset," *The International Journal of Robotics Research*, vol. 32, no. 11, pp. 1231–1237, 2017, doi: 10.1177/0278364913491297. [Online]. Available: https://doi.org/10.1177/0278364913491297

[13] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele, "The cityscapes dataset for semantic urban scene understanding," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 3213–3223.

[14] 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.

[15] O. Sener and S. Savarese, "Active learning for convolutional neural networks: A core-set approach," *arXiv preprint arXiv:1708.00489*, 2017.

[16] H. Yin and C. Berger, "When to use what data set for your self-driving car algorithm: An overview of publicly available driving datasets," in *2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 2017, pp. 1–8.

[17] M. Liu, E. Yurtsever, J. Fossaert, X. Zhou, W. Zimmer, Y. Cui, B. L. Zagar, and A. C. Knoll, "A survey on autonomous driving datasets: Statistics, annotation quality, and a future outlook," *IEEE Transactions on Intelligent Vehicles*, pp. 1–29, 2024.

[18] Y. Li, J. Yang, and J. Wen, "Entropy-based redundancy analysis and information screening," *Digital Communications and Networks*, vol. 9, no. 5, pp. 1061–1069, 2023.

[19] B. Sorscher, R. Geirhos, S. Shekhar, S. Ganguli, and A. Morcos, "Beyond neural scaling laws: beating power law scaling via data pruning," *Advances in Neural Information Processing Systems*, vol. 35, pp. 19 523–19 536, 2022.

[20] H. Yu, X. Yang, S. Zheng, and C. Sun, "Active learning from imbalanced data: A solution of online weighted extreme learning machine," *IEEE transactions on neural networks and learning systems*, vol. 30, no. 4, pp. 1088–1103, 2018.

[21] D. Cohn, L. Atlas, and R. Ladner, "Improving generalization with active learning," *Machine Learning*, vol. 15, pp. 201–221, 1994.

[22] P. Ren, Y. Xiao, X. Chang, P.-Y. Huang, Z. Li, B. B. Gupta, X. Chen, and X. Wang, "A survey of deep active learning," *ACM computing surveys (CSUR)*, vol. 54, no. 9, pp. 1–40, 2021.

[23] D. Zhao, H. Lam, H. Peng, S. Bao, D. J. LeBlanc, K. Nobukawa, and C. S. Pan, "Accelerated evaluation of automated vehicles safety in lane-change scenarios based on importance sampling techniques," *IEEE Transactions on Intelligent Transportation Systems*, vol. 18, no. 3, pp. 595–607, Marc 2017.

[24] D. Zhao, X. Huang, H. Peng, H. Lam, and D. J. LeBlanc, "Accelerated evaluation of automated vehicles in car-following maneuvers," *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, no. 3, pp. 733–744, Marc 2018.

[25] M. Noshad, J. Choi, Y. Sun, A. Hero III, and I. D. Dinov, "A data value metric for quantifying information content and utility," *Journal of big Data*, vol. 8, no. 1, p. 82, 2021.

[26] X. Wang, Y. Chen, and W. Zhu, "A survey on curriculum learning," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 9, pp. 4555–4576, 2021.

[27] T. M. Cover and J. A. Thomas, *Elements of Information Theory*, 2nd ed. John Wiley & Sons, 2006.

[28] A. Krause and C. Guestrin, "Near-optimal observation selection using submodular functions," in *AAAI*, vol. 7, 2007, pp. 1650–1654.

[29] S. P. Borgatti, "Centrality and network flow," *Social networks*, vol. 27, no. 1, pp. 55–71, 2005.

[30] M. E. Newman, "Modularity and community structure in networks," *Proceedings of the national academy of sciences*, vol. 103, no. 23, pp. 8577–8582, 2006.

[31] G. Jocher and J. Qiu, "Ultralytics yolo11," 2024. [Online]. Available: https://github.com/ultralytics/ultralytics

[32] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo, "Swin transformer: Hierarchical vision transformer using shifted windows," *arXiv preprint arXiv:2103.14030*, 2021.

[33] K. Chen, J. Wang, J. Pang, Y. Cao, Y. Xiong, X. Li, S. Sun, W. Feng, Z. Liu, J. Xu, Z. Zhang, D. Cheng, C. Zhu, T. Cheng, Q. Zhao, B. Li, X. Lu, R. Zhu, Y. Wu, J. Dai, J. Wang, J. Shi, W. Ouyang, C. C. Loy, and D. Lin, "MMDetection: Open mmlab detection toolbox and benchmark," *arXiv preprint arXiv:1906.07155*, 2019.

[34] L.-C. Chen, G. Papandreou, F. Schroff, and H. Adam, "Rethinking atrous convolution for semantic image segmentation," *arXiv preprint arXiv:1706.05587*, 2017.

[35] M. Contributors, "MMSegmentation: Openmmlab semantic segmentation toolbox and benchmark," https://github.com/open-mmlab/mmsegmentation, 2020.

[36] A. H. Lang, S. Vora, H. Caesar, L. Zhou, J. Yang, and O. Beijbom, "Pointpillars: Fast encoders for object detection from point clouds," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 12 697–12 705.

[37] M. Contributors, "MMDetection3D: OpenMMLab next-generation platform for general 3D object detection," https://github.com/open-mmlab/mmdetection3d, 2020.

[38] J.-R. Chang and Y.-S. Chen, "Pyramid stereo matching network," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 5410–5418.

[39] M. Liang, B. Yang, R. Hu, Y. Chen, R. Liao, S. Feng, and R. Urtasun, "Learning lane graph representations for motion forecasting," in *ECCV*, 2020.

[40] Z. Huang, J. Gu, J. Li, and X. Yu, "A stereo matching algorithm based on the improved psmnet," *Plos one*, vol. 16, no. 8, p. e0251657, 2021.