SynQuE: Estimating Synthetic Dataset Quality Without Annotations

Anonymous authors
Paper under double-blind review

Abstract

We introduce and formalize the **Synthetic Dataset Quality Estimation** (SynQuE) problem: ranking synthetic datasets by their expected real-world task performance using only limited unannotated real data. This addresses a critical and open challenge where data is scarce due to collection costs or privacy constraints. We establish the first comprehensive benchmarks for this problem by introducing and evaluating proxy metrics that choose synthetic data for training to maximize task performance on real data. We introduce the first proxy metrics for SynQuE by adapting distribution and diversity-based distance measures to our context via embedding models. To address the shortcomings of these metrics on complex planning tasks, we propose Lens, a novel proxy that leverages large language model reasoning. Our results show that SynQuE proxies correlate with real task performance across diverse tasks, including sentiment analysis, Text2SQL, web navigation, and image classification, with Lens consistently outperforming others on complex tasks by capturing nuanced characteristics. For instance, on text-to-SQL parsing, training on the top-3 synthetic datasets selected via SynQuE proxies can raise accuracy from 30.4% to 38.4 (+8.1)% on average compared to selecting data indiscriminately. This work establishes SynQuE as a practical framework for synthetic data selection under real-data scarcity and motivates future research on foundation model-based data characterization and fine-grained data selection.

1 Introduction

Data scarcity hinders effective machine learning, especially for tasks requiring specialized expertise like autonomous navigation or natural language interfaces, where data collection is costly and slow (Xie et al., 2024; Yang et al., 2024). In sensitive domains such as healthcare and finance (Tan et al., 2024; Jordan & Mitchell, 2015), privacy concerns further complicate data acquisition. Large generative models have emerged as capable synthetic data generators, producing annotated data for tasks like policy learning (Xu et al., 2024), Text2SQL (Yang et al., 2024), sentiment analysis (Ye et al., 2022; Li et al., 2023c), and image classification (Geng et al., 2025). While synthetic data can improve real-world performance under scarcity, results vary widely depending on task and data quality (Huang et al., 2025; Geng et al., 2025).

Can we distinguish between high-quality synthetic data that improves real-world task performance and low-quality data that offers little benefit, *without any annotated real data* and *without costly model training*? Crucially, increasing the size of synthetic datasets does not always lead to better downstream performance as it does with real data; in some cases, larger synthetic datasets can even degrade performance, exhibiting inverse scaling trends (Geng et al., 2025; Li et al., 2023c; Setlur et al., 2024; Gao et al., 2022; Møller et al., 2023). Therefore, selecting a synthetic dataset from a pool of datasets to train on to optimize downstream performance is important.

We introduce **Syn**thetic Dataset **Qu**ality **E**stimation, or SynQuE, the problem of ranking multiple synthetic datasets by quality using only limited unannotated samples of real data. A synthetic dataset A is of higher quality than B if a model trained on A outperforms one trained on B on a real-world test set. This ability is crucial when real data annotation is costly or infeasible. For example, in text-to-SQL parsing, SynQuE helps select the synthetic dataset that yields better generalization from a small set of unannotated real queries.

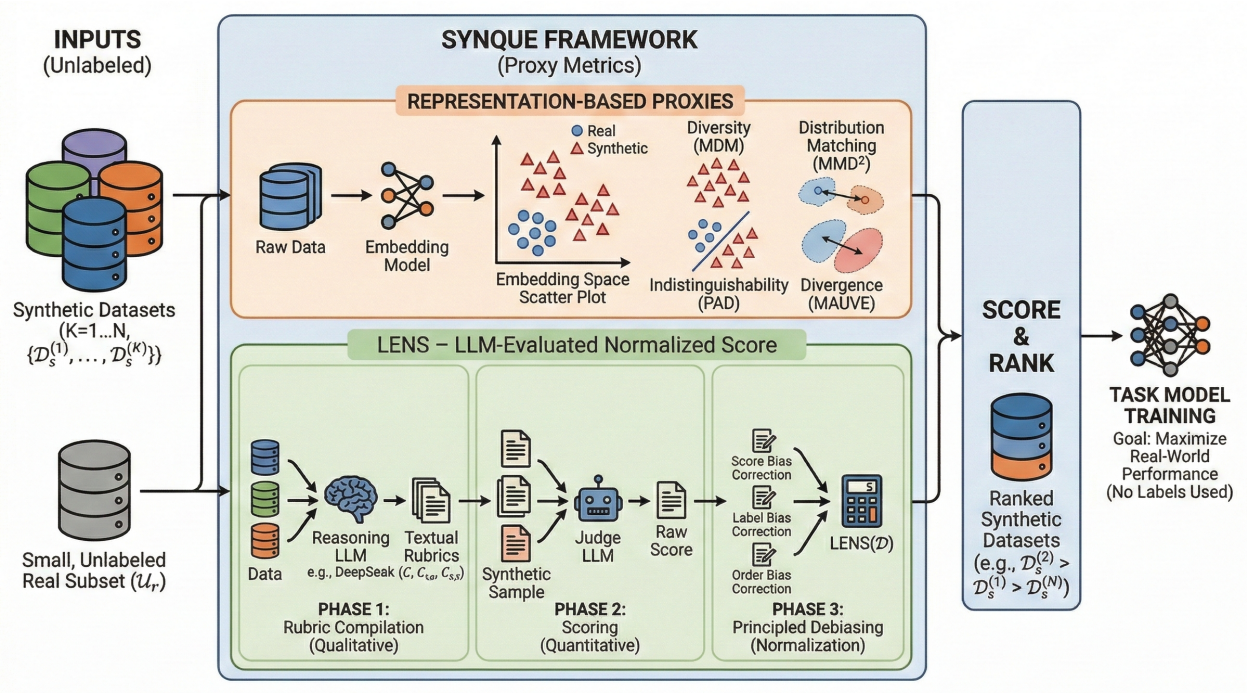


Figure 1: SynQuE uses synthetic data and unlabeled samples of real data to estimate synthetic data quality. Proxy scores are used to rank and select the datasets that lead to the best task performance.

Similarly, for intelligent web agents, it identifies the synthetic interactions that produce agents performing best on real navigation tasks.

This work makes three main contributions. 1) We formalize the SynQuE problem and establish the first comprehensive benchmark. As part of this, we introduce and evaluate a suite of *proxy metrics*: computable scores that estimate a synthetic dataset's quality using only a subset of unannotated real data. We adapt proxy metrics using established distributional measures such as mean distance to medoids (Cox et al., 2021), diversity measures such as Proxy-A-Distance (Ben-David et al., 2006), and divergence measures such as MAUVE (Pillutla et al., 2021) — none have been systematically evaluated for the purpose of synthetic data selection. 2) We propose **L**LM-**E**valuated **N**ormalized **S**core (Lens), a novel proxy measure that leverages large language model (LLM) reasoning to create *dataset rubrics* that highlight difference between synthetic data and real data in language. 3) We conduct a comprehensive experiment across diverse domains in sentiment analysis, Text2SQL, web navigation, and image classification in order to evaluate how well these proxy metrics are able to select synthetic data to maximize performance on real test data.

Our empirical evaluation shows that SynQuE proxies exhibit moderate to strong correlation with real task performance across diverse domains including sentiment analysis, Text-to-SQL, image classification, and web navigation. While the best proxy varies by task, most can effectively predict downstream performance without *any* labeled real data, enabling practical synthetic data selection that outperforms indiscriminate synthetic data selection. We find that the reliability of proxies depends on task complexity, with higher variance on noisy data like synthetic images. Among the proxies, Lens, leveraging LLM reasoning and a principled debiasing strategy, consistently achieves superior gains on complex tasks like web navigation by capturing nuanced task details beyond embedding-based metrics. These results establish SynQuE as a robust framework for selecting high-quality synthetic data and motivate future work on stronger foundation model methods to characterize data and perform fine-grained, example-level data selection.

2 Related Work

Data synthesis Synthesizing training data with generative models is a promising way to address data scarcity by leveraging instruction-following abilities (Touvron et al., 2023; Ouyang et al., 2022) and vast pre-trained knowledge (Long et al., 2024; Honovich et al., 2022; Mishra et al., 2022). This has been applied

across domains such as text classification (Ye et al., 2022), Text-to-SQL (Yang et al., 2024; Lei et al., 2024; Li et al., 2024), planning (Sun et al., 2024; Hu et al., 2024; Xu et al., 2024; Murty et al., 2025), and computer vision (Geng et al., 2025; Li et al., 2022). While synthetic data often improves downstream models (Liu et al., 2024; Ye et al., 2022), challenges remain in ensuring quality due to issues such as hallucination (Huang et al., 2025), mode collapse (Goodfellow et al., 2014; Durall et al., 2020; Shumailov et al., 2024), and counterfactual artifacts (Li et al., 2023c; Yu et al., 2023). Our work introduces the SynQuE framework for selecting synthetic data to maximize task performance without access to large amounts of real data and without training task models.

Distributional and diversity metrics Metrics like Proxy-$\mathcal{A}$-Distance (**PAD**) estimate domain divergence by training classifiers to distinguish source and target data (Ben-David et al., 2010; 2006; Quinonero-Candela et al., 2022). While effective in some NLP tasks (Elsahar & Gallé, 2019), **PAD** struggles in noisy, complex settings such as agent planning (He et al., 2024). Diversity metrics like mean-distance-to-medoids (**MDM**)(Cox et al., 2021) and DCScore(Zhu et al., 2025) approximate the coverage of synthetic data. However, these measures do not capture common synthetic data factual inconsistencies from generative model hallucinations (Huang et al., 2025; Li et al., 2023b; Gunjal et al., 2023), which can harm downstream performance (Geng et al., 2025; Yu et al., 2023; Casco-Rodriguez et al., 2023; Li et al., 2023c; Hataya et al., 2023; Briesch et al., 2023).

Data selection and weighting A growing body of work focuses on selecting high-value training samples. Some approaches utilize scaling laws by training regression models on short runs to predict the best data mixture (Liu et al., 2025; Magnusson et al., 2025). Others employ influence functions or gradient-based heuristics to select influential subsets (Zhang et al., 2024; Hu et al., 2025), or utilize small real datasets to learn instance-level weights for synthetic data (Kuo et al., 2025). Crucially, these methods focus on *instance-level* selection or weighting, often requiring access to labeled validation sets to compute gradients or train selection models. In contrast, SynQuE addresses *dataset-level* ranking in a zero-shot, gradient-free manner, making it suitable for "cold-start" settings where no labeled data exists. Finally, another group of works try to distinguish between human and machine text using divergence measures (Pillutla et al., 2021) or LLM-as-a-judge (Gu et al., 2025; Krumdick et al., 2025; Zheng et al., 2023). Neither of these two techniques have been studied for synthetic data selection. Furthermore, the latter also suffers from limitations of generative models previously mentioned such as hallucination. Our work establishes the first systematic benchmark and results for synthetic data selection in a practical, fully unsupervised regime.

Dataset Ranking and Selection Most prior work has focused on data selection at the sample-level, aiming to curate optimal subsets of individual data points to reduce training time or labeling costs. For example, Kaushal et al. (2018) leverage submodular functions to maximize diversity and coverage in data subsets, while active learning frameworks like BatchBALD (Kirsch et al., 2019) acquire batches of informative points based on Bayesian mutual information. However, ranking or selecting at the dataset-level is equally important, as it accounts for dependencies and structure among samples within a dataset.

Recent methodologies have expanded selection to this macro scale. For instance, Nadas et al. (2025) introduce a hierarchical framework designed to retrieve entire real-world datasets from massive, decentralized sources under strict computational constraints. Similarly, previous work on dataset retrieval, such as (Chapman et al., 2020), studies data set ranking using keyword-based queries and defines data set quality primarily in terms of search relevance.

In contrast to both instance-level subset curation and the macro-level routing or retrieval of real data, our setting addresses the emerging challenge of selecting *synthetic* datasets under *label scarcity*. We are concerned with ranking candidate model-generated datasets according to their potential downstream utility when annotated validation data is unavailable. Rather than optimizing search relevance or structural efficiency across data silos, our notion of quality relies on unannotated target data to estimate real-world task performance – a fundamentally different objective.

3 The Synthetic Dataset Quality Estimation Problem

We define the SynQuE problem and establish notation. Let $\mathcal{D}_\mathrm{r} = \{x_\mathrm{r}^{(i)}, y_\mathrm{r}^{(i)}\}_{i=1}^{n_\mathrm{r}}$ be a real dataset, which we assume is scarce and for which labels are unavailable for validation. Instead, we only have access to a small, unannotated collection of real-world inputs $\mathcal{U}_\mathrm{r} = \{x_i\}_{i=1}^{m_\mathrm{r}} \in \mathcal{D}_\mathrm{r}$. Our goal is to use $\mathcal{U}_\mathrm{r}$ to estimate the quality of K synthetic datasets $\{\mathcal{D}_\mathrm{s}^{(1)}, \ldots, \mathcal{D}_\mathrm{s}^{(K)}\}$. These datasets might be generated by different methods and thus vary in quality. We aim to select the synthetic dataset $\mathcal{D}_\mathrm{s}^*$ that yields the best model performance on $\mathcal{D}_\mathrm{r}$, *without using labeled real data or training models on any synthetic dataset.*

Consider a model $f(\cdot;\theta_k)$ trained on the synthetic dataset $\mathcal{D}_\mathrm{s}^{(k)}$ with parameters θ_k. Let $M(f, \mathcal{D}_e)$ denote model f's performance on the evaluation dataset $\mathcal{D}_e$, for instance task completion rate or accuracy. The ideal synthetic dataset $\mathcal{D}_\mathrm{s}^*$ satisfies:

$$\mathcal{D}_\mathrm{s}^* = \mathrm{argmax}_{\mathcal{D}_\mathrm{s}^{(k)}} M\left(f\left(\cdot;\theta_k\right), \mathcal{D}_e\right) \tag{1}$$

In practice, Eq 1 is infeasible because it requires labeled real data for evaluation and extensive model training across K synthetic datasets. Instead, SynQuE seeks *proxy metrics* $Q(\mathcal{D}_\mathrm{s}^{(k)}, \mathcal{U}_\mathrm{r})$—computable using only the synthetic dataset $\mathcal{D}_\mathrm{s}^{(k)}$ and unannotated real samples $\mathcal{U}_\mathrm{r}$—that correlate strongly with true downstream performance. The SynQuE problem thus reduces to designing or learning a proxy function $Q : (\mathcal{D}_\mathrm{s}^{(k)}, \mathcal{U}_\mathrm{r}) \to \mathbb{R}$ such that a higher score indicates the synthetic dataset $\mathcal{D}_\mathrm{s}^{(k)}$ is more likely to produce better real-world task performance. This formulation enables efficient and effective use of synthetic data in low-resource or privacy-sensitive scenarios. For ease of exposition, we refer to the score produced by the proxy function Q as the **SynQuE score**.

4 SynQuE Proxy Metrics

In this section, we introduce the proxy metrics designed for SynQuE. To avoid circular reasoning, we explicitly ground these proxies in machine learning theory, linking them directly to our downstream objective in eq. (1) through two primary indicators: **distribution matching** and **diversity**.

We ground our similarity metrics (**PAD**, **MMD**^{2}) in domain adaptation theory (Ben-David et al., 2006). Theoretical bounds show that downstream target risk is upper-bounded by source training risk plus domain divergence. Therefore, ranking synthetic datasets by their divergence from real data mathematically targets the upper bound of the downstream error defined in eq. (1).

Similarly, we ground our diversity proxy (**MDM**) in embedding space coverage. If a synthetic dataset lacks diversity (e.g., due to mode collapse), its empirical support is artificially restricted. Models trained on this limited support fail to generalize to the broader real-world target distribution, directly increasing target risk.

We adapt these traditional distance and diversity measures for SynQuE by embedding raw text or image data into continuous representations. Finally, for complex, long-horizon tasks where representation-based metrics fail, we introduce Lens. This LLM-based measure operates directly on raw data, providing deep contextual understanding without relying on a universal embedding model.

4.1 Representation-Based Proxy Metrics

Mean Distance to Medoid Our first proxy metric adapts Mean Distance to Medoid (**MDM**), a measure of dataset diversity (Cox et al., 2021; Laliberté & Legendre, 2010; Lehman & Stanley, 2011; Risi et al., 2009). The rationale is that high-diversity synthetic datasets may offer broader coverage in the embedding space and thus yield higher performance on real data. For SynQuE, **MDM** characterizes this diversity by measuring the sparsity of data points around medoids. Given $\mathcal{D}_\mathrm{s}$ of size n_s, we compute K medoids using clustering algorithms such as kMedoids [1]. Let $\tilde{x}_k$ be the k-th medoid and $\mathcal{C}_k$ be the set of points assigned to this medoid. We aggregate the Euclidean distances from all points within their corresponding cluster: $\mathbf{MDM} = \frac{1}{n_s}\sum_{k=1}^{K}\sum_{x_i \in \mathcal{C}_k} d(x_i, \tilde{x}_k)$. Intuitively, if points within each cluster are centered around the medoid, **MDM** will be small, indicating the dataset is less diverse in the embedding space. In contrast, high-diversity

[1] `https://pypi.org/project/kmedoids/`

Listing 1: Rubric prompt

```
You are shown samples from
datasets A and B. Give up to
10 points describing how
dataset B is similar to dataset

Samples from dataset A:
${location A}

Samples from dataset B:
${location B}
```

Listing 2: Scorer Prompt

```
Given similarities and differences between
datasets, how likely is the given sample from
dataset ${prediction}? Choose from very unlikely,
unlikely, unsure, likely, and very likely.

Similarities: ${similarities}

Differences: ${differences}

Sample: ${sample}
```

Table 1: Simplified LENS prompt templates. Appendix C contains detailed prompts for each task.

synthetic datasets to have broader coverage, and therefore higher performance on real data. A higher **MDM** score suggests greater diversity, so we use it directly as the SYNQUE score.

Maximum Mean Discrepancy Next, we propose a proxy based on Maximum Mean Discrepancy ($\mathbf{MMD}^2$), a nonparametric test that assesses whether two samples originate from the same distribution (Gretton et al., 2012; Borgwardt et al., 2006; Lu et al., 2022; Li et al., 2015). As a SYNQUE proxy, we use $\mathbf{MMD}^2$ to measure the discrepancy between the synthetic dataset and the real data distribution. Given n_s synthetic input samples $x_i \in \mathcal{D}_\mathrm{s}$ of size n_s and m_r unannotated real input samples $y_i \in \mathcal{U}_\mathrm{r}$, $\mathbf{MMD}^2$ quantifies the distance between these two empirical distributions in a reproducing kernel Hilbert space using a kernel function $k(\cdot)$.

$$\mathbf{MMD}^2 = \frac{1}{{m_\mathrm{r}}^2} \sum_{i,j=1}^{m_\mathrm{r}} k(x_i, x_j) + \frac{1}{{n_\mathrm{s}}^2} \sum_{i,j=1}^{n_\mathrm{s}} k(y_i, y_j) - \frac{2}{m_\mathrm{r} n_\mathrm{s}} \sum_{i,j=1}^{n_\mathrm{s}, m_\mathrm{r}} k(x_i, y_j) \tag{2}$$

A smaller $\mathbf{MMD}^2$ score indicates the synthetic data distribution is closer to the real one. To maintain consistency with other proxies where higher is better, we use $-\mathbf{MMD}^2$ as the SYNQUE score.

Proxy-A-Distance Our third representation-based proxy adapts Proxy-A-Distance (**PAD**), a discriminative measure from domain adaptation that quantifies the divergence between two distributions (Ben-David et al., 2006; Elsahar & Gallé, 2019). The **PAD** proxy measures how well a classifier can discriminate between samples from the synthetic dataset $\mathcal{D}_\mathrm{s}$ and real dataset $\mathcal{D}_\mathrm{r}$. We compute $\mathbf{PAD} = 1 - 2\mathcal{E}(G)$ by training a binary domain classifier $G : x \to [0, 1]$ (e.g., a linear SVM, a multi-layer perceptron) to distinguish between synthetic and real inputs (i.e. we do not assume access to labels), where the error of the classifier $\mathcal{E}$ can be computed as:

$$\mathcal{E}(G) = 1 - \frac{1}{n_\mathrm{s} + m_\mathrm{r}} \sum_{x_i \in \mathcal{D}_\mathrm{s}, \mathcal{U}_\mathrm{r}} \left| G(x_i) - \mathbb{I}(x_i \in \mathcal{D}_\mathrm{s}) \right| \tag{3}$$

A higher classification error implies the datasets are not easily separable, indicating lower divergence and thus higher synthetic data quality. For consistency with other divergence metrics, use $-\mathbf{PAD}$ as the SYNQUE score.

Mauve As an established generative baseline to compare against our representation-based proxies, we evaluate MAUVE (Pillutla et al., 2021), a widely used metric that quantifies the divergence between *text* distributions. MAUVE summarizes both Type I and Type II errors. Given a synthetic data distribution Q and a real data distribution P, Type I error means Q generates text that is unlikely under P (unrealistic samples), while Type II error means Q fails to generate text that is plausible under P (lacks diversity). MAUVE captures both error types in a single score, which approaches 1 as the distributions become more similar. Since a higher score indicates better alignment with real data, we utilize the MAUVE score directly as a baseline dataset ranking metric to compare against our proposed SYNQUE proxies.

4.2 LLM-Evaluated Normalized Score (Lens)

The representation-based proxies described so far rely on high-quality continuous representations of inputs. In low-resource settings where such representations are unavailable, or in long-horizon settings where it is intractable to compress a long sequence of observations and states into a compact, fixed-size representation, these representation-based proxies may fall short. To address this, we introduce **L**LM-**E**valuated **N**ormalized **S**core (Lens), a novel method that leverages LLMs as zero-shot discriminators. Lens first derives a language **rubric** describing the similarities and differences between samples of unannotated real data $\mathcal{U}_{\mathrm{r}}$ and inputs from the synthetic dataset $\mathcal{D}_{\mathrm{s}}$. A subsequent (potentially smaller) LLM then scores how likely each synthetic example is to belong to the real dataset, guided by the rubric. The average score across the synthetic dataset is used as the final SynQuE score. The intuition is similar to that behind **PAD**: a higher classification error by the rubric-guided scorer implies higher synthetic data quality. We now detail how Lens is computed.

Rubric compilation Given real input samples $\mathcal{U}_{\mathrm{r}}$, we collect an equal number of samples $\mathcal{U}_{\mathrm{s}}$ from the synthetic dataset $\mathcal{D}_{\mathrm{s}}$. Both collections are given to a reasoning LLM (e.g. `DeepSeek R1` or `o4-mini`) to generate three sets of **characteristic descriptions**: commonalities (C), differences of real from synthetic ($C_{\mathrm{r,s}}$), and differences of synthetic from real ($C_{\mathrm{s,r}}$). Listing 1 shows a simplified rubric compilation prompt template. Our design is backed by the principled idea of approximating domain divergence through discriminator error (Ben-David et al., 2006): Lens's scoring is motivated by **PAD**, where the error of a classifier (here, the LLM-based scorer) reflects the distance between distributions. Unlike **PAD**, however, Lens does not require a pretrained encoder to map samples into fixed-length representations; instead, it operates directly on the native data format and characterizes differences using language rubrics.

Principled Debiasing and Scoring We now describe how to compute the score of a synthetic dataset. A key challenge of scoring is in mitigating LLM biases. We identified three primary sources:

1. **Order Bias:** The set of differences an LLM derives when comparing A to B can differ significantly from when comparing B to A.
2. **Label Bias:** When asked how likely an example x belongs to A or B, an LLM may score both as "very likely", a contradiction.
3. **Score Bias:** LLMs may have an inherent preference for certain score values (e.g., "likely") regardless of the input.

To address these systematically, we employ a *minimal design* involving four scoring permutations for each sample. Specifically, we evaluate the likelihood of each sample belonging to the target dataset (real or synthetic) under both difference rubrics ($C_{\mathrm{s,r}}$ and $C_{\mathrm{r,s}}$). This yields four distinct scores *per sample* x: $g_{\mathrm{r}|C_{\mathrm{s,r}}}(x)$, $g_{\mathrm{s}|C_{\mathrm{s,r}}}(x)$, $g_{\mathrm{r}|C_{\mathrm{r,s}}}(x)$, $g_{\mathrm{s}|C_{\mathrm{r,s}}}(x)$.

We formalize the LLM scoring function as $g_{\mathcal{D}|C} : \mathcal{X} \to \{0, 1, 2, 3, 4\}$, where $x \in \mathcal{X}$ denotes a sample from the input space. The output score (0-4) indicates how likely the example x belongs to dataset $\mathcal{D}$ given characteristic descriptions C in natural language.

First, to mitigate **score bias**, we compute baseline scores by averaging the LLM's judgments on real inputs $x \in \mathcal{U}_{\mathrm{r}}$ for each of the four permutations. For instance, the baseline for scoring an example as real, given the description of how synthetic differs from real, is:

$$z_{\mathrm{r}|C_{\mathrm{s,r}}} = \mathbb{E}\left[g_{\mathrm{r}|C_{\mathrm{s,r}}}(x)\right] \approx \frac{1}{n_{\mathrm{r}}} \sum_{i=1}^{n_{\mathrm{r}}} g_{\mathrm{r}|C_{\mathrm{s,r}}}(x_i) \tag{4}$$

We then compute a *score-debiased* score h for each synthetic sample by normalizing its raw score against this baseline:

$$h_{\mathrm{r}|C_{\mathrm{s,r}}}(x) = \frac{g_{\mathrm{r}|C_{\mathrm{s,r}}}(x)}{\max(\epsilon, z_{\mathrm{r}|C_{\mathrm{s,r}}})} \tag{5}$$

Here, ϵ is a small constant to avoid division by zero. Intuitively, this scores-debiased score expresses how much the LLM scores the example compared to how it usually scores real examples. Similarly, the score-debiased score for the synthetic label is computed as:

$$h_{\mathrm{s}|C_{\mathrm{s,r}}}(x) = \frac{g_{\mathrm{s}|C_{\mathrm{s,r}}}(x)}{\max(\epsilon, z_{\mathrm{s}|C_{\mathrm{s,r}}})} \tag{6}$$

where $z_{\mathrm{s}|C_{\mathrm{s,r}}} = \frac{1}{n_{\mathrm{r}}} \sum_{i=1}^{n_{\mathrm{r}}} g_{\mathrm{s}|C_{\mathrm{s,r}}}(x_i)$. Next, to compute a **label-debiased** score, we normalize the LLM's preference for the "real" label over the "synthetic" label for each synthetic example:

$$p_{\mathrm{r}|C_{\mathrm{s,r}}}(x) = \frac{h_{\mathrm{r}|C_{\mathrm{s,r}}}(x)}{h_{\mathrm{r}|C_{\mathrm{s,r}}}(x) + h_{\mathrm{s}|C_{\mathrm{s,r}}}(x) + \epsilon} \tag{7}$$

Finally, to create an **order-debiased** error score for a synthetic sample $x \in \mathcal{D}_{\mathrm{s}}$, we average the label-debiased scores indicating a "real" prediction using both sets of difference descriptions ($C_{\mathrm{s,r}}$ and $C_{\mathrm{r,s}}$):

$$\hat{p}_{\mathrm{s}}(x) = \frac{1}{2}\left[p_{\mathrm{r}|C_{\mathrm{s,r}}}(x) + p_{\mathrm{r}|C_{\mathrm{r,s}}}(x)\right] \tag{8}$$

Conversely, we compute a symmetrical error score for each real sample $x \in \mathcal{U}_{\mathrm{r}}$, representing the likelihood it is misclassified as synthetic. Using corresponding baselines normalized against the synthetic dataset, this is calculated as:

$$\hat{p}_{\mathrm{r}}(x) = \frac{1}{2}\left[p_{\mathrm{s}|C_{\mathrm{s,r}}}(x) + p_{\mathrm{s}|C_{\mathrm{r,s}}}(x)\right] \tag{9}$$

The final Lens score is the combined, weighted empirical mean of these classification errors across both datasets, capturing the overall indistinguishability between the real and synthetic samples:

$$\textsc{Lens}(\mathcal{D}_{\mathrm{s}}) = \frac{1}{n_{\mathrm{s}} + n_{\mathrm{r}}}\left(\sum_{i=1}^{n_{\mathrm{s}}} \hat{p}_{\mathrm{s}}(x_i) + \sum_{j=1}^{n_{\mathrm{r}}} \hat{p}_{\mathrm{r}}(x_j)\right) \tag{10}$$

5 Experiments

We choose four diverse tasks spanning different machine learning domains to examine how well each candidate proxy metric extrapolates to real data performance on tasks with varying complexities and modalities. For Lens, we incorporate `Deepseek-R1` to generate 10 points about similar and different characteristics $C_{\mathrm{s,r}}$ between synthetic data samples $\mathcal{U}_{\mathrm{s}}$ and real data samples $\mathcal{U}_{\mathrm{r}}$. We then use `Qwen2.5-32B-Instruct` (Qwen et al., 2025) and `8B` to score synthetic examples according to the rubric. For image domain, we use OpenAI `o4-mini` to compile rubrics and `Qwen2.5-VL-32B-Instruct` to score. For representation-based metrics **PAD**, **MDM**, and **MMD**^{2}, we use state-of-the-art `qte-Qwen2-7B-Instruct` to embed text inputs and `E5-V` (Jiang et al., 2024) to embed image inputs for proxy scoring. We use XGBoost (Chen & Guestrin, 2016) to compute **PAD** and polynomial kernel for **MMD**^{2} (Gretton et al., 2012). We include additional kernel ablations in Appendix 11. We use the official release[2] from Mauve, with the default hyperparameter setting for Mauve calculation.

We also include an experiment with perplexity-based metric as additional baseline in Text2SQL (see table 5). Perplexity, inspired by scaling law methods (Magnusson et al., 2025; Liu et al., 2025), fine-tunes a model on each synthetic data set and measures its perplexity on the unannotated real data subset; a lower perplexity is expected to indicate higher quality.

We use Pearson (Pearson & Galton, 1997) and Spearman rank (Spearman, 1904) correlation coefficients to measure how strongly task performance and proxy scores are related. Pearson focuses on *predictability* by capturing linear relationships, while Spearman focuses on *trend* by evaluating whether the relationship between variables is consistently increasing or decreasing (i.e., monotonic), regardless of the exact shape. To reduce variance in correlation analysis across different sample subsets, for all tasks, we construct subsets $\mathcal{U}_{\mathrm{r}}$ by sampling with five different seeds. Final correlation scores are averaged across seeds.

[2] `https://pypi.org/project/mauve-text/`

Table 2: Top-3 task performance for all candidate proxies of SynQuE. Top-3 task performance is computed by averaging task performance of synthetic datasets chosen using each proxy metric. Improvements are calculated based on increase over average performance of all synthetic datasets.

		Top-3 Ranked Average Task Performance across Proxy Metrics							
Tasks	Test mean	Lens 7B debiased	Lens 7B biased	Lens 32B debiased	Lens 32B biased	**PAD**	**MMD^2**	**MDM**	**Mauve**
Sentiment	49.6	50.5 +0.8	51.2 +1.6	52.0 +2.4	51.0 +1.4	55.3 +5.7	54.7 +5.1	54.2 +4.6	54.6 +4.9
Text2SQL									
Computer	45.8	46.6 +0.7	46.4 +0.6	48.3 +2.5	46.3 +0.5	48.3 +2.5	47.4 +1.6	48.2 +2.3	48.3 +2.5
Apps	30.4	34.7 +4.4	36.3 +5.9	33.8 +3.4	35.2 +4.9	33.5 +3.2	38.4 +8.1	33.9 +3.5	38.4 +8.1
Movies	37.3	41.0 +3.7	41.4 +4.1	43.8 +6.5	44.7 +7.4	44.2 +6.9	43.0 +5.7	46.9 +9.6	44.6 +7.3
Average	37.8	40.8 +2.9	41.4 +3.5	42.0 +4.1	42.1 +4.3	42.0 +4.2	42.9 +5.1	43.0 +5.2	43.8 +6.0
Image									
Split 1	57.2	57.3 +0.2	56.7 -0.4	56.4 -0.8	53.4 -3.7	55.9 -1.3	57.3 +0.1	57.0 -0.1	56.0 -1.1
Split 2	55.8	55.3 -0.4	55.8 +0.0	56.2 +0.4	56.0 +0.2	55.4 -0.4	54.5 -1.3	54.8 -1.0	56.3 +0.5
Split 3	57.7	59.1 +1.4	58.7 +1.0	60.2 +2.5	57.0 -0.7	58.2 +0.6	64.1 +6.5	52.2 -5.5	58.4 +0.7
Average	56.9	57.2 +0.4	57.0 +0.2	57.6 +0.7	55.5 -1.4	56.5 -0.4	58.6 +1.8	54.7 -2.2	56.9 +0
WebNav	25.8	26.5 +0.7	26.3 +0.5	26.3 +0.5	26.0 +0.2	25.7 -0.1	26.5 +0.7	25.8 -0.1	26.3 +0.5

Sentiment Analysis Recent work shows LLMs overfit widely-used datasets due to data contamination (Balloccu et al., 2024; Sainz et al., 2023; Oren et al., 2023). To mitigate this, we evaluate on a domain-specific financial tweets sentiment dataset[3]. We create 32 synthetic class-balanced datasets (998 samples each) using eight prompt types: zero-shot, zero-shot with background knowledge, with train-time or test-time stock ticker info, and few-shot variants. We use `Qwen2.5-7B-Instruct`, `Qwen2.5-32B-Instruct`, `Llama3.1-8B-Instruct`, and `Llama3.3-70B-Instruct` (Grattafiori et al., 2024) for each prompt type. Background knowledge uses detailed guideline instructions for better task alignment. Stock tickers are sampled one at a time for synthesis. Details are in section C. We train task models using XGBoost and evaluate F1 score on a 2,388-item test set. Rubrics are compiled by randomly sampling 200 points from each real and synthetic dataset.

Text2SQL We evaluate SynQuE on Text2SQL using three DBs from the BIRD benchmark (Movies, App Store, Computer Students — we denote the last two as Apps and Computers) (Li et al., 2023a). We synthesize 1,000 data points with 4 prompt types: zero-shot with background knowledge (guidelines and schema), zero-shot with test-time info (random table rows), and few-shot (three examples). `Qwen2.5-7B-Instruct` and `Llama3.1-8B-Instruct` models are used for dataset generation. Task models are finetuned from `Qwen2.5-Coder-1.5B-Instruct` following CodeS[4] and evaluated using execution accuracy on the real test set. For rubrics, we sample 30 points per synthetic and real dataset. Real data sizes are 60, 69, and 164 for Apps, Computers, and Movies respectively.

Image Classification In addition to text-only settings, we evaluate SynQuE on image classification using synthetic datasets curated from `unmet-promise`[5]. These datasets are created with different prompts using Stable Diffusion 1.1 and 1.5: label, label plus physical relation, and label plus background description (Geng et al., 2025). Images are mapped to ImageNet classes (Deng et al., 2009) via caption analysis C, then filtered with a vision-language model to remove noisy labels. Data cleaning details are provided in Appendix section C. The final set includes 15 classes with 300 images each. Due to limited samples, the 15-class task is split into three 5-class tasks (table 4). We train ResNet-50 (He et al., 2015) from scratch for 50 epochs, early stopping on 10% validation data. Evaluation uses mean reciprocal rank (MRR) for finer performance measurement.

[3] `https://huggingface.co/datasets/zeroshot/twitter-financial-news-sentiment`
[4] `https://github.com/RUCKBReasoning/codes`
[5] `https://huggingface.co/datasets/scottgeng00/unmet-promise`

Table 3: Spearman (left) and Pearson (right) correlation scores of SynQuE proxy metrics. Lens uses a fraction of samples for rubric compilation except for Web Navigation tasks.

Tasks	Lens 7B				Lens 32B				**PAD**		**MMD^2**		**MDM**		Mauve	
	debiased		biased		debiased		biased									
Sentiment	.25	.33	.26	.17	.38	.26	.24	.23	.53	.65	.45	.67	.68	.85	.53	.57
Text2SQL																
Computer	.19	.13	.10	.10	.41	.45	.18	.23	.46	.69	.33	.85	.39	.63	.24	.78
Apps	.38	.37	.42	.49	.46	.40	.55	.61	.43	.42	.53	.79	.44	.56	.74	.52
Movies	.41	.50	.41	.26	.50	.46	.56	.47	.50	.64	.38	.46	.61	.41	.65	.68
Average	.33	.33	.31	.28	.46	.43	.43	.44	.46	.58	.41	.70	.48	.53	.55	.66
Image																
Split 1	-.18	-.19	-.30	-.35	-.28	-.28	-.68	-.67	-.06	-.05	.66	.52	-.37	-.27	-.04	-.20
Split 2	.02	-.04	.14	.05	.20	.05	.20	.05	.20	.31	.09	.17	.03	-.32	.03	.13
Split 3	-.10	-.01	-.15	-.03	.31	.33	.37	.44	.02	-.15	.26	.21	-.54	-.76	.46	.34
Average	-.09	-.08	-.10	-.11	.08	.03	-.04	-.06	.05	.04	.33	.30	-.30	-.45	.15	.09
WebNav	.15	.17	.11	.18	.15	.15	.08	.09	.11	.08	-.02	.06	-.11	-.08	-.09	-.10

Rubrics are constructed from 100 sampled images per real and synthetic data, consistent across SynQuE methods.

Web Navigation Our fourth task evaluates SynQuE on agentic web navigation planning using WebVoyager (He et al., 2024) and synthetic data from NNetNav (Murty et al., 2025). Inputs include task objectives, current step observations (accessibility tree), and past actions; targets are actions leading to task success. WebVoyager has 15 websites; we exclude Google Flights and Booking which are no longer feasible (Zhou et al., 2024; Murty et al., 2025), leaving 13 sites with 557 tasks. Each site forms a test domain split into 5 synthetic subsets. Models are fine-tuned with LoRA (Hu et al., 2021) on `Qwen2.5-7B-Instruct`. We use all synthetic and 20 real samples per method.

Table 4: Division of the 15 selected ImageNet classes into three 5-class splits for image classification tasks. Each row corresponds to one split used in our experiments.

Splits	ImageNet classes
1	bra, mask, lion, cloak, tank
2	hammer, backpack, stage, throne, tray
3	plate, desk, kimono, shield, church

5.1 Results Analysis

SynQuE proxies correlate with task performance and improve selection. table 3 shows that SynQuE proxy metrics demonstrate moderate to strong correlation with downstream task performance. To show the practical utility of these proxies, we simulate selecting the top 3 datasets based on each metric's score and compare their average task performance against the mean performance of all available synthetic datasets. As shown in table 2, nearly all proxy metrics significantly improve dataset selection over selecting synthetic datasets non-discriminately (i.e. uniform selection). This demonstrates that SynQuE is an effective framework for maximizing real-data performance, despite not having access to labeled real data and only a limited sample of real data. We also conduct an experiment with Perplexity on Text2SQL, to examine the effectiveness as a potential proxy. As shown in table 5, it correlates poorly with even text data, therefore we conclude that scaling methods would not work under our setting, where no annotated data is available for evaluation to build a regression model that predicts the best data mixture.

Table 5: Spearman (left) and Pearson (right) correlations with Perplexity scoring on the BIRD Text2SQL benchmark

	Perplexity	
Computers	-.31	-.33
Apps	-.25	-.29
Movies	.24	.31
Average	-.11	-.10

Table 6: Spearman (left) and Pearson (right) correlations with different number of real samples for scoring Text2SQL. LENS uses debiased 32B scoring. Note that results in table 3 use 30 real samples. "Comp." denotes Computers.

	$\lvert\mathcal{U}_\mathrm{r}\rvert = 25$										$\lvert\mathcal{U}_\mathrm{r}\rvert = 50$									
	Lens		**PAD**		**MMD^2**		**MDM**		**Mauve**		**Lens**		**PAD**		**MMD^2**		**MDM**		**Mauve**	
Comp.	.22	.22	.36	.65	.28	.80	.39	.65	.19	.65	.64	.38	.51	.78	.34	.87	.39	.62	.87	.78
Apps	.29	.48	-.20	-.23	.40	.77	.44	.56	.65	.70	.64	.66	.68	.76	.57	.80	.44	.56	.32	.65
Movies	.33	.48	.33	.36	.52	.57	.60	.40	-.14	-.35	.43	.49	.57	.37	.81	.56	.67	.45	.81	.60
Average	.28	.39	.16	.26	.40	.71	.47	.54	.23	.33	.57	.51	.59	.64	.57	.74	.50	.55	.67	.68

Performance on ambiguous image data shows high variance. The synthetic image classification data contains significant visual variability and label ambiguity, especially in Split 2 between classes like "stage" and "throne" (fig. 2a, fig. 2b). This confuses most proxy metrics, resulting in inconsistent correlations across splits (table 3). However, table 2 shows that when used for selection, several proxies (e.g. debiased LENS 32B, **MMD^2**) still improve average task performance.

Using more real samples improves correlation. As shown in table 6, increasing the number of unannotated real samples m_r consistently leads to stronger correlations for all proxy metrics. This indicates that even a modest increase in available real-world data can significantly improve the reliability of synthetic data quality estimation.

Lens excels on complex, long-horizon tasks. As shown in both tables, the 32B debiased LENS is the only proxy that consistently achieves positive correlation and improves top-3 task performance across all tasks and splits. Its advantage is particularly pronounced in web navigation, a complex planning task where representation-based metrics struggle. LENS leverages LLM reasoning over rich, structured inputs like accessibility trees to generate interpretable rubrics. For instance, an example characteristic point for the website "Wolfram Alpha" is: "Dataset B tasks focus on data retrieval (e.g. temperature anomalies, moon phases) while Dataset A emphasizes applied computational problem-solving". These specific nuances in long text (e.g. instructions, state observations) are difficult to capture using general-purpose dense vector embedders, which explains why LENS outperforms representation-based proxy metrics on complex, abstract tasks like web navigation. This method does exhibit weaker correlation in image classification. We hypothesize that this stems from the inability of VLMs to capture meaning characteristics descriptions in batches of images during rubric generation, and VLM rubric generation will likely improve as VLMs improve in quality.

5.2 Ablation Studies and Cost Analysis

Cost-Effectiveness of SynQuE Proxies A key motivation for SYNQUE is efficiency. Our representation-based proxies require a one-time embedding computation, after which scoring all datasets is nearly instantaneous (e.g., 19 seconds for MMD on 32 datasets). LENS, using modern LLM serving frameworks, is also highly efficient, taking ~ 15 seconds per dataset with a 32B model on a H200 GPU. In contrast, perplexity-based data selection, inspired by scaling-law studies (Liu et al., 2025), require training many (e.g., 512 1M models used in their experiment) small models on the mixture of all synthetic datasets, a significantly more costly procedure, yet yield weaker correlations (table 3). This highlights the practical advantage of the SYNQUE framework.

Larger scorers lead to stronger correlations We find that larger scoring models yield stronger correlations between LENS and task performance, as shown in table 3. Intuitively, this is expected because larger models generally possess more robust instruction-following and reasoning capabilities, enabling them to better assess data quality and align proxy scores with downstream performance.

Lens is robust to preferential bias in LLM training data. To address concerns that an LLM evaluator might favor data it generated, we tested LENS with different scoring models on data generated by `Qwen2.5`.

The results, detailed in Appendix table 7, show that performance is consistent across evaluators, including those distinct from models used to generate the synthetic data, indicating minimal preferential bias.

Principled debiasing and rubrics are critical for Lens. As illustrated in table 3, the correlations between Lens and task performance consistently increase when debiasing is applied, indicating that raw scores may be systematically biased and do not reliably reflect true data quality. Once debiasing is introduced, the correlation becomes strongly positive and consistently outperforms the biased scores. This demonstrates that debiasing effectively corrects for these systematic errors and aligns Lens scores with actual task performance. Further ablations show that using a rubric consistently improves correlation over a zero-shot baseline (Appendix table 9), and that 10 rubric points generally offer the best trade-off between specificity and generality (Appendix table 10).

6 Practitioner Guidelines

Based on our empirical evaluation including ablation studies across diverse domains and experiment setups, we offer the following guidelines for practitioners utilizing the SYNQUE framework to select synthetic data.

- **Simple, One-shot NLP Tasks (e.g., Sentiment Analysis):** Representation-based metrics like **PAD** and **MDM** serve as effective and *computationally efficient* defaults. These metrics require only a one-time embedding computation and demonstrate remarkable robustness, maintaining strong correlations with task performance even with as few as 100 real samples.
- **Complex, Long-horizon Tasks (e.g., Agentic, Large-Scale Datasets):** We highly recommend utilizing Lens. General-purpose dense vector embedders often struggle to capture the specific nuances and sequential logic present in long texts like state observations; Lens overcomes this by leveraging LLM reasoning to generate interpretable rubrics over structured inputs.
- **Vision Tasks:** Selecting high-quality data is particularly challenging due to significant visual variability and label ambiguity. In these settings, distribution matching via $\mathbf{MMD}^2$ provides a solid baseline that can still improve dataset selection, though it should be paired with explicit data cleaning pipelines – such as utilizing vision-language models to filter out noisy or hallucinated images.
- **Safeguarding Against Generative Issues:** To safeguard against common generative issues like mode collapse, **MDM** should be utilized to measure dataset diversity. If a synthetic dataset lacks diversity, its empirical support is artificially restricted; therefore, prioritizing higher MDM scores ensures broader coverage in the embedding space and better generalization to the real-world target distribution.

7 Conclusions, Limitations and Future Work

We formalized the SynQuE problem of ranking synthetic datasets by their impact on real-world task performance using limited unannotated real data. Our comprehensive evaluation established that various proxies can reliably predict downstream performance, offering a cost-effective alternative to full model training. We proposed Lens, a novel proxy leveraging LLM reasoning and principled debiasing, which consistently outperforms others on challenging, long-horizon tasks. Overall, SynQuE offers a robust framework for synthetic data selection when labeled real data is scarce.

SynQuE assumes that real data is scarce, a setting not all deployments face. While Lens performs well on the complex tasks studied, its effectiveness should be validated on more diverse tasks. Additionally, we experiment with limited-size LLMs due to resource constraints. Future work should explore 1) scaling Lens to larger sizes and different architectures, especially strong VLMs, to assess generality and improvements. 2) using rubric feedback to guide LLMs in synthesizing more realistic data, and 3) developing fine-grained, example-level proxy use to directly improve task model training.

8 Broader Impact Statement

While utilizing synthetic datasets offers a promising solution to data scarcity, it could introduce ethical considerations regarding the potential amplification of biases. It is crucial to clarify that our SynQuE framework does not select datasets by directly optimizing for downstream test performance, as such labels are strictly inaccessible during the selection phase. Instead, proxies such as **PAD**, **MMD**^{2}, and Mauve evaluate synthetic data based strictly on its **unsupervised distributional similarity** to a small real-world reference set; test performance is utilized solely as a post-hoc oracle to evaluate the success of these proxies. Consequently, the risk of amplifying known biases depends entirely on the composition of this real-world anchor set. If the reference data contains historical or representational biases, similarity-based selection will inherently prioritize synthetic datasets that reflect and perpetuate those same biases. To mitigate this risk, we strongly advise practitioners to rigorously audit their real-world reference samples for representational fairness and equity prior to applying SynQuE, ensuring that the anchor data aligns with desired ethical standards.

References

Simone Balloccu, Patrícia Schmidtová, Mateusz Lango, and Ondřej Dušek. Leak, Cheat, Repeat: Data Contamination and Evaluation Malpractices in Closed-Source LLMs. 2024. doi: 10.48550/ARXIV.2402.03927. URL https://arxiv.org/abs/2402.03927. Version Number: 2.

Shai Ben-David, John Blitzer, Koby Crammer, and Fernando Pereira. Analysis of Representations for Domain Adaptation. In *Advances in Neural Information Processing Systems*, volume 19. MIT Press, 2006. URL https://proceedings.neurips.cc/paper_files/paper/2006/hash/b1b0432ceafb0ce714426e9114852ac7-Abstract.html.

Shai Ben-David, John Blitzer, Koby Crammer, Alex Kulesza, Fernando Pereira, and Jennifer Wortman Vaughan. A theory of learning from different domains. *Machine Learning*, 79(1):151–175, May 2010. ISSN 1573-0565. doi: 10.1007/s10994-009-5152-4. URL https://doi.org/10.1007/s10994-009-5152-4.

Karsten M. Borgwardt, Arthur Gretton, Malte J. Rasch, Hans-Peter Kriegel, Bernhard Schölkopf, and Alex J. Smola. Integrating structured biological data by Kernel Maximum Mean Discrepancy. *Bioinformatics*, 22(14):e49–e57, July 2006. ISSN 1367-4811, 1367-4803. doi: 10.1093/bioinformatics/btl242. URL https://academic.oup.com/bioinformatics/article/22/14/e49/228383.

Martin Briesch, Dominik Sobania, and Franz Rothlauf. Large Language Models Suffer From Their Own Output: An Analysis of the Self-Consuming Training Loop. 2023. doi: 10.48550/ARXIV.2311.16822. URL https://arxiv.org/abs/2311.16822. Version Number: 2.

Josue Casco-Rodriguez, Sina Alemohammad, Lorenzo Luzi, Ahmed Imtiaz, Hossein Babaei, Daniel LeJeune, Ali Siahkoohi, and Richard Baraniuk. Self-Consuming Generative Models go MAD. *LatinX in AI at Neural Information Processing Systems Conference 2023*, December 2023. doi: 10.52591/lxai202312101. URL https://research.latinxinai.org/papers/neurips/2023/pdf/Josue_CascoRodriguez.pdf. Conference Name: LatinX in AI at Neural Information Processing Systems Conference 2023.

Adriane Chapman, Elena Simperl, Laura Koesten, George Konstantinidis, Luis-Daniel Ibáñez, Emilia Kacprzak, and Paul Groth. Dataset search: a survey. *The VLDB Journal*, 29(1):251–272, January 2020. ISSN 0949-877X. doi: 10.1007/s00778-019-00564-x. URL https://doi.org/10.1007/s00778-019-00564-x.

Tianqi Chen and Carlos Guestrin. XGBoost: A Scalable Tree Boosting System. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 785–794, August 2016. doi: 10.1145/2939672.2939785. URL http://arxiv.org/abs/1603.02754. arXiv:1603.02754 [cs].

Samuel Rhys Cox, Yunlong Wang, Ashraf Abdul, Christian Von Der Weth, and Brian Y. Lim. Directed Diversity: Leveraging Language Embedding Distances for Collective Creativity in Crowd Ideation. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, pp. 1–35, Yokohama Japan, May 2021. ACM. ISBN 978-1-4503-8096-6. doi: 10.1145/3411764.3445782. URL https://dl.acm.org/doi/10.1145/3411764.3445782.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 248–255, June 2009. doi: 10.1109/CVPR.2009.5206848. URL https://ieeexplore.ieee.org/document/5206848.

Ricard Durall, Avraam Chatzimichailidis, Peter Labus, and Janis Keuper. Combating Mode Collapse in GAN training: An Empirical Analysis using Hessian Eigenvalues, December 2020. URL http://arxiv.org/abs/2012.09673. arXiv:2012.09673 [cs].

Hady Elsahar and Matthias Gallé. To Annotate or Not? Predicting Performance Drop under Domain Shift. In Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan (eds.), *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pp. 2163–2173, Hong Kong, China, November 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1222. URL https://aclanthology.org/D19-1222/.

Jiahui Gao, Renjie Pi, Yong Lin, Hang Xu, Jiacheng Ye, Zhiyong Wu, Weizhong Zhang, Xiaodan Liang, Zhenguo Li, and Lingpeng Kong. Self-Guided Noise-Free Data Generation for Efficient Zero-Shot Learning. May 2022. URL https://www.semanticscholar.org/paper/Self-Guided-Noise-Free-Data-Generation-for-Learning-Gao-Pi/6f7e03e4ccd26c762090e25dc5d2eb1e1f8c641d.

Scott Geng, Cheng-Yu Hsieh, Vivek Ramanujan, Matthew Wallingford, Chun-Liang Li, Pang Wei Koh, and Ranjay Krishna. The Unmet Promise of Synthetic Training Images: Using Retrieved Real Images Performs Better, January 2025. URL http://arxiv.org/abs/2406.05184. arXiv:2406.05184 [cs].

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative Adversarial Nets. In *Advances in Neural Information Processing Systems*, volume 27. Curran Associates, Inc., 2014. URL https://proceedings.neurips.cc/paper_files/paper/2014/hash/5ca3e9b122f61f8f06494c97b1afccf3-Abstract.html.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, Arun Rao, Aston Zhang, Aurelien Rodriguez, Austen Gregerson, Ava Spataru, Baptiste Roziere, Bethany Biron, Binh Tang, Bobbie Chern, Charlotte Caucheteux, Chaya Nayak, Chloe Bi, Chris Marra, Chris McConnell, Christian Keller, Christophe Touret, Chunyang Wu, Corinne Wong, Cristian Canton Ferrer, Cyrus Nikolaidis, Damien Allonsius, Daniel Song, Danielle Pintz, Danny Livshits, Danny Wyatt, David Esiobu, Dhruv Choudhary, Dhruv Mahajan, Diego Garcia-Olano, Diego Perino, Dieuwke Hupkes, Egor Lakomkin, Ehab AlBadawy, Elina Lobanova, Emily Dinan, Eric Michael Smith, Filip Radenovic, Francisco Guzmán, Frank Zhang, Gabriel Synnaeve, Gabrielle Lee, Georgia Lewis Anderson, Govind Thattai, Graeme Nail, Gregoire Mialon, Guan Pang, Guillem Cucurell, Hailey Nguyen, Hannah Korevaar, Hu Xu, Hugo Touvron, Iliyan Zarov, Imanol Arrieta Ibarra, Isabel Kloumann, Ishan Misra, Ivan Evtimov, Jack Zhang, Jade Copet, Jaewon Lee, Jan Geffert, Jana Vranes, Jason Park, Jay Mahadeokar, Jeet Shah, Jelmer van der Linde, Jennifer Billock, Jenny Hong, Jenya Lee, Jeremy Fu, Jianfeng Chi, Jianyu Huang, Jiawen Liu, Jie Wang, Jiecao Yu, Joanna Bitton, Joe Spisak, Jongsoo Park, Joseph Rocca, Joshua Johnstun, Joshua Saxe, Junteng Jia, Kalyan Vasuden Alwala, Karthik Prasad, Kartikeya Upasani, Kate Plawiak, Ke Li, Kenneth Heafield, Kevin Stone, Khalid El-Arini, Krithika Iyer, Kshitiz Malik, Kuenley Chiu, Kunal Bhalla, Kushal Lakhotia, Lauren Rantala-Yeary, Laurens van der Maaten, Lawrence Chen, Liang Tan, Liz Jenkins, Louis Martin, Lovish Madaan, Lubo Malo, Lukas Blecher, Lukas Landzaat, Luke de Oliveira, Madeline Muzzi, Mahesh Pasupuleti, Mannat Singh, Manohar Paluri, Marcin Kardas, Maria Tsimpoukelli, Mathew Oldham, Mathieu Rita, Maya Pavlova, Melanie Kambadur, Mike Lewis, Min Si, Mitesh Kumar Singh, Mona Hassan, Naman Goyal, Narjes Torabi, Nikolay Bashlykov, Nikolay Bogoychev, Niladri Chatterji, Ning Zhang, Olivier Duchenne, Onur Çelebi, Patrick Alrassy, Pengchuan Zhang, Pengwei Li, Petar Vasic, Peter Weng, Prajjwal Bhargava, Pratik Dubal, Praveen Krishnan, Punit Singh Koura, Puxin Xu, Qing He, Qingxiao Dong, Ragavan Srinivasan, Raj Ganapathy, Ramon Calderer, Ricardo Silveira Cabral, Robert Stojnic, Roberta Raileanu, Rohan Maheswari, Rohit Girdhar, Rohit Patel, Romain Sauvestre, Ronnie Polidoro, Roshan Sumbaly, Ross Taylor, Ruan Silva, Rui Hou, Rui Wang, Saghar Hosseini, Sahana Chennabasappa,

Sanjay Singh, Sean Bell, Seohyun Sonia Kim, Sergey Edunov, Shaoliang Nie, Sharan Narang, Sharath Raparthy, Sheng Shen, Shengye Wan, Shruti Bhosale, Shun Zhang, Simon Vandenhende, Soumya Batra, Spencer Whitman, Sten Sootla, Stephane Collot, Suchin Gururangan, Sydney Borodinsky, Tamar Herman, Tara Fowler, Tarek Sheasha, Thomas Georgiou, Thomas Scialom, Tobias Speckbacher, Todor Mihaylov, Tong Xiao, Ujjwal Karn, Vedanuj Goswami, Vibhor Gupta, Vignesh Ramanathan, Viktor Kerkez, Vincent Gonguet, Virginie Do, Vish Vogeti, Vítor Albiero, Vladan Petrovic, Weiwei Chu, Wenhan Xiong, Wenyin Fu, Whitney Meers, Xavier Martinet, Xiaodong Wang, Xiaofang Wang, Xiaoqing Ellen Tan, Xide Xia, Xinfeng Xie, Xuchao Jia, Xuewei Wang, Yaelle Goldschlag, Yashesh Gaur, Yasmine Babaei, Yi Wen, Yiwen Song, Yuchen Zhang, Yue Li, Yuning Mao, Zacharie Delpierre Coudert, Zheng Yan, Zhengxing Chen, Zoe Papakipos, Aaditya Singh, Aayushi Srivastava, Abha Jain, Adam Kelsey, Adam Shajnfeld, Adithya Gangidi, Adolfo Victoria, Ahuva Goldstand, Ajay Menon, Ajay Sharma, Alex Boesenberg, Alexei Baevski, Allie Feinstein, Amanda Kallet, Amit Sangani, Amos Teo, Anam Yunus, Andrei Lupu, Andres Alvarado, Andrew Caples, Andrew Gu, Andrew Ho, Andrew Poulton, Andrew Ryan, Ankit Ramchandani, Annie Dong, Annie Franco, Anuj Goyal, Aparajita Saraf, Arkabandhu Chowdhury, Ashley Gabriel, Ashwin Bharambe, Assaf Eisenman, Azadeh Yazdan, Beau James, Ben Maurer, Benjamin Leonhardi, Bernie Huang, Beth Loyd, Beto De Paola, Bhargavi Paranjape, Bing Liu, Bo Wu, Boyu Ni, Braden Hancock, Bram Wasti, Brandon Spence, Brani Stojkovic, Brian Gamido, Britt Montalvo, Carl Parker, Carly Burton, Catalina Mejia, Ce Liu, Changhan Wang, Changkyu Kim, Chao Zhou, Chester Hu, Ching-Hsiang Chu, Chris Cai, Chris Tindal, Christoph Feichtenhofer, Cynthia Gao, Damon Civin, Dana Beaty, Daniel Kreymer, Daniel Li, David Adkins, David Xu, Davide Testuggine, Delia David, Devi Parikh, Diana Liskovich, Didem Foss, Dingkang Wang, Duc Le, Dustin Holland, Edward Dowling, Eissa Jamil, Elaine Montgomery, Eleonora Presani, Emily Hahn, Emily Wood, Eric-Tuan Le, Erik Brinkman, Esteban Arcaute, Evan Dunbar, Evan Smothers, Fei Sun, Felix Kreuk, Feng Tian, Filippos Kokkinos, Firat Ozgenel, Francesco Caggioni, Frank Kanayet, Frank Seide, Gabriela Medina Florez, Gabriella Schwarz, Gada Badeer, Georgia Swee, Gil Halpern, Grant Herman, Grigory Sizov, Guangyi, Zhang, Guna Lakshminarayanan, Hakan Inan, Hamid Shojanazeri, Han Zou, Hannah Wang, Hanwen Zha, Haroun Habeeb, Harrison Rudolph, Helen Suk, Henry Aspegren, Hunter Goldman, Hongyuan Zhan, Ibrahim Damlaj, Igor Molybog, Igor Tufanov, Ilias Leontiadis, Irina-Elena Veliche, Itai Gat, Jake Weissman, James Geboski, James Kohli, Janice Lam, Japhet Asher, Jean-Baptiste Gaya, Jeff Marcus, Jeff Tang, Jennifer Chan, Jenny Zhen, Jeremy Reizenstein, Jeremy Teboul, Jessica Zhong, Jian Jin, Jingyi Yang, Joe Cummings, Jon Carvill, Jon Shepard, Jonathan McPhie, Jonathan Torres, Josh Ginsburg, Junjie Wang, Kai Wu, Kam Hou U, Karan Saxena, Kartikay Khandelwal, Katayoun Zand, Kathy Matosich, Kaushik Veeraraghavan, Kelly Michelena, Keqian Li, Kiran Jagadeesh, Kun Huang, Kunal Chawla, Kyle Huang, Lailin Chen, Lakshya Garg, Lavender A, Leandro Silva, Lee Bell, Lei Zhang, Liangpeng Guo, Licheng Yu, Liron Moshkovich, Luca Wehrstedt, Madian Khabsa, Manav Avalani, Manish Bhatt, Martynas Mankus, Matan Hasson, Matthew Lennie, Matthias Reso, Maxim Groshev, Maxim Naumov, Maya Lathi, Meghan Keneally, Miao Liu, Michael L. Seltzer, Michal Valko, Michelle Restrepo, Mihir Patel, Mik Vyatskov, Mikayel Samvelyan, Mike Clark, Mike Macey, Mike Wang, Miquel Jubert Hermoso, Mo Metanat, Mohammad Rastegari, Munish Bansal, Nandhini Santhanam, Natascha Parks, Natasha White, Navyata Bawa, Nayan Singhal, Nick Egebo, Nicolas Usunier, Nikhil Mehta, Nikolay Pavlovich Laptev, Ning Dong, Norman Cheng, Oleg Chernoguz, Olivia Hart, Omkar Salpekar, Ozlem Kalinli, Parkin Kent, Parth Parekh, Paul Saab, Pavan Balaji, Pedro Rittner, Philip Bontrager, Pierre Roux, Piotr Dollar, Polina Zvyagina, Prashant Ratanchandani, Pritish Yuvraj, Qian Liang, Rachad Alao, Rachel Rodriguez, Rafi Ayub, Raghotham Murthy, Raghu Nayani, Rahul Mitra, Rangaprabhu Parthasarathy, Raymond Li, Rebekkah Hogan, Robin Battey, Rocky Wang, Russ Howes, Ruty Rinott, Sachin Mehta, Sachin Siby, Sai Jayesh Bondu, Samyak Datta, Sara Chugh, Sara Hunt, Sargun Dhillon, Sasha Sidorov, Satadru Pan, Saurabh Mahajan, Saurabh Verma, Seiji Yamamoto, Sharadh Ramaswamy, Shaun Lindsay, Shaun Lindsay, Sheng Feng, Shenghao Lin, Shengxin Cindy Zha, Shishir Patil, Shiva Shankar, Shuqiang Zhang, Shuqiang Zhang, Sinong Wang, Sneha Agarwal, Soji Sajuyigbe, Soumith Chintala, Stephanie Max, Stephen Chen, Steve Kehoe, Steve Satterfield, Sudarshan Govindaprasad, Sumit Gupta, Summer Deng, Sungmin Cho, Sunny Virk, Suraj Subramanian, Sy Choudhury, Sydney Goldman, Tal Remez, Tamar Glaser, Tamara Best, Thilo Koehler, Thomas Robinson, Tianhe Li, Tianjun Zhang, Tim Matthews, Timothy Chou, Tzook Shaked, Varun Vontimitta, Victoria Ajayi, Victoria Montanez, Vijai Mohan, Vinay Satish Kumar, Vishal Mangla, Vlad Ionescu, Vlad Poenaru, Vlad Tiberiu Mihailescu, Vladimir Ivanov, Wei Li, Wenchen Wang, Wenwen Jiang, Wes Bouaziz, Will Constable, Xiaocheng Tang, Xiaojian Wu, Xiaolan Wang, Xilun Wu,

Xinbo Gao, Yaniv Kleinman, Yanjun Chen, Ye Hu, Ye Jia, Ye Qi, Yenda Li, Yilin Zhang, Ying Zhang, Yossi Adi, Youngjin Nam, Yu, Wang, Yu Zhao, Yuchen Hao, Yundi Qian, Yunlu Li, Yuzi He, Zach Rait, Zachary DeVito, Zef Rosnbrick, Zhaoduo Wen, Zhenyu Yang, Zhiwei Zhao, and Zhiyu Ma. The Llama 3 Herd of Models, November 2024. URL http://arxiv.org/abs/2407.21783. arXiv:2407.21783 [cs].

Arthur Gretton, Karsten M. Borgwardt, Malte J. Rasch, Bernhard Schölkopf, and Alexander Smola. A Kernel Two-Sample Test. *Journal of Machine Learning Research*, 13(25):723–773, 2012. ISSN 1533-7928. URL http://jmlr.org/papers/v13/gretton12a.html.

Jiawei Gu, Xuhui Jiang, Zhichao Shi, Hexiang Tan, Xuehao Zhai, Chengjin Xu, Wei Li, Yinghan Shen, Shengjie Ma, Honghao Liu, Saizhuo Wang, Kun Zhang, Yuanzhuo Wang, Wen Gao, Lionel Ni, and Jian Guo. A Survey on LLM-as-a-Judge, March 2025. URL http://arxiv.org/abs/2411.15594. arXiv:2411.15594 [cs].

Anisha Gunjal, Jihan Yin, and Erhan Bas. Detecting and Preventing Hallucinations in Large Vision Language Models. arXiv, 2023. doi: 10.48550/ARXIV.2308.06394. URL https://arxiv.org/abs/2308.06394. Version Number: 3.

Ryuichiro Hataya, Han Bao, and Hiromi Arai. Will Large-scale Generative Models Corrupt Future Datasets? *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 20498–20508, October 2023. doi: 10.1109/ICCV51070.2023.01879. URL https://ieeexplore.ieee.org/document/10376575/. Conference Name: 2023 IEEE/CVF International Conference on Computer Vision (ICCV) ISBN: 9798350307184 Place: Paris, France.

Hongliang He, Wenlin Yao, Kaixin Ma, Wenhao Yu, Yong Dai, Hongming Zhang, Zhenzhong Lan, and Dong Yu. WebVoyager: Building an End-to-End Web Agent with Large Multimodal Models, June 2024. URL http://arxiv.org/abs/2401.13919. arXiv:2401.13919 [cs].

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep Residual Learning for Image Recognition, December 2015. URL http://arxiv.org/abs/1512.03385. arXiv:1512.03385 [cs].

Or Honovich, Thomas Scialom, Omer Levy, and Timo Schick. Unnatural Instructions: Tuning Language Models with (Almost) No Human Labor, December 2022. URL http://arxiv.org/abs/2212.09689. arXiv:2212.09689 [cs].

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-Rank Adaptation of Large Language Models, October 2021. URL http://arxiv.org/abs/2106.09685. arXiv:2106.09685 [cs].

Mengkang Hu, Pu Zhao, Can Xu, Qingfeng Sun, Jianguang Lou, Qingwei Lin, Ping Luo, and Saravan Rajmohan. AgentGen: Enhancing Planning Abilities for Large Language Model based Agent via Environment and Task Generation, November 2024. URL http://arxiv.org/abs/2408.00764. arXiv:2408.00764 [cs].

Yuzheng Hu, Pingbang Hu, Han Zhao, and Jiaqi W. Ma. Most Influential Subset Selection: Challenges, Promises, and Beyond, January 2025. URL http://arxiv.org/abs/2409.18153. arXiv:2409.18153 [cs].

Lei Huang, Weijiang Yu, Weitao Ma, Weihong Zhong, Zhangyin Feng, Haotian Wang, Qianglong Chen, Weihua Peng, Xiaocheng Feng, Bing Qin, and Ting Liu. A Survey on Hallucination in Large Language Models: Principles, Taxonomy, Challenges, and Open Questions. *ACM Transactions on Information Systems*, 43(2):1–55, March 2025. ISSN 1046-8188, 1558-2868. doi: 10.1145/3703155. URL https://dl.acm.org/doi/10.1145/3703155.

Ting Jiang, Minghui Song, Zihan Zhang, Haizhen Huang, Weiwei Deng, Feng Sun, Qi Zhang, Deqing Wang, and Fuzhen Zhuang. E5-V: Universal Embeddings with Multimodal Large Language Models, July 2024. URL http://arxiv.org/abs/2407.12580. arXiv:2407.12580 [cs].

M. I. Jordan and T. M. Mitchell. Machine learning: Trends, perspectives, and prospects. *Science*, 349 (6245):255–260, July 2015. doi: 10.1126/science.aaa8415. URL https://www.science.org/doi/10.1126/science.aaa8415.

Vishal Kaushal, Anurag Sahoo, Khoshrav Doctor, Narasimha Raju, Suyash Shetty, Pankaj Singh, Rishabh Iyer, and Ganesh Ramakrishnan. Learning From Less Data: Diversified Subset Selection and Active Learning in Image Classification Tasks, May 2018. URL http://arxiv.org/abs/1805.11191. arXiv:1805.11191 [cs].

Andreas Kirsch, Joost van Amersfoort, and Yarin Gal. BatchBALD: Efficient and Diverse Batch Acquisition for Deep Bayesian Active Learning. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. URL https://proceedings.neurips.cc/paper_files/paper/2019/hash/95323660ed2124450caaac2c46b5ed90-Abstract.html.

Michael Krumdick, Charles Lovering, Varshini Reddy, Seth Ebner, and Chris Tanner. No Free Labels: Limitations of LLM-as-a-Judge Without Human Grounding, March 2025. URL http://arxiv.org/abs/2503.05061. arXiv:2503.05061 [cs].

Hsun-Yu Kuo, Yin-Hsiang Liao, Yu-Chieh Chao, Wei-Yun Ma, and Pu-Jen Cheng. Not All LLM-Generated Data Are Equal: Rethinking Data Weighting in Text Classification, March 2025. URL http://arxiv.org/abs/2410.21526. arXiv:2410.21526 [cs].

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. Efficient Memory Management for Large Language Model Serving with PagedAttention, September 2023. URL http://arxiv.org/abs/2309.06180. arXiv:2309.06180 [cs].

Etienne Laliberté and Pierre Legendre. A distance-based framework for measuring functional diversity from multiple traits. *Ecology*, 91(1):299–305, 2010. ISSN 1939-9170. doi: 10.1890/08-2244.1. URL https://onlinelibrary.wiley.com/doi/abs/10.1890/08-2244.1. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1890/08-2244.1.

Joel Lehman and Kenneth O. Stanley. Abandoning Objectives: Evolution Through the Search for Novelty Alone. *Evolutionary Computation*, 19(2):189–223, June 2011. ISSN 1063-6560. doi: 10.1162/EVCO_a_00025. URL https://ieeexplore.ieee.org/abstract/document/6793380.

Fangyu Lei, Jixuan Chen, Yuxiao Ye, Ruisheng Cao, Dongchan Shin, Hongjin Su, Zhaoqing Suo, Hongcheng Gao, Wenjing Hu, Pengcheng Yin, Victor Zhong, Caiming Xiong, Ruoxi Sun, Qian Liu, Sida Wang, and Tao Yu. Spider 2.0: Evaluating Language Models on Real-World Enterprise Text-to-SQL Workflows, November 2024. URL http://arxiv.org/abs/2411.07763. arXiv:2411.07763 [cs].

Daiqing Li, Huan Ling, Seung Wook Kim, Karsten Kreis, Sanja Fidler, and Antonio Torralba. BigDatasetGAN: Synthesizing ImageNet with Pixel-wise Annotations. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 21298–21308, New Orleans, LA, USA, June 2022. IEEE. ISBN 978-1-6654-6946-3. doi: 10.1109/CVPR52688.2022.02064. URL https://ieeexplore.ieee.org/document/9878775/.

Haoyang Li, Jing Zhang, Hanbing Liu, Ju Fan, Xiaokang Zhang, Jun Zhu, Renjie Wei, Hongyan Pan, Cuiping Li, and Hong Chen. CodeS: Towards Building Open-source Language Models for Text-to-SQL, February 2024. URL http://arxiv.org/abs/2402.16347. arXiv:2402.16347.

Jinyang Li, Binyuan Hui, Ge Qu, Jiaxi Yang, Binhua Li, Bowen Li, Bailin Wang, Bowen Qin, Rongyu Cao, Ruiying Geng, Nan Huo, Xuanhe Zhou, Chenhao Ma, Guoliang Li, Kevin C. C. Chang, Fei Huang, Reynold Cheng, and Yongbin Li. Can LLM Already Serve as A Database Interface? A BIg Bench for Large-Scale Database Grounded Text-to-SQLs, November 2023a. URL http://arxiv.org/abs/2305.03111. arXiv:2305.03111 [cs].

Yifan Li, Yifan Du, Kun Zhou, Jinpeng Wang, Wayne Xin Zhao, and Ji-Rong Wen. Evaluating Object Hallucination in Large Vision-Language Models. arXiv, 2023b. doi: 10.48550/ARXIV.2305.10355. URL https://arxiv.org/abs/2305.10355. Version Number: 3.

Yujia Li, Kevin Swersky, and R. Zemel. Generative Moment Matching Networks. February 2015. URL https://www.semanticscholar.org/paper/2904a9932f4cd0f0886121dc1f2d4aaac0455176.

Zhuoyan Li, Hangxiao Zhu, Zhuoran Lu, and Ming Yin. Synthetic Data Generation with Large Language Models for Text Classification: Potential and Limitations. 2023c. doi: 10.48550/ARXIV.2310.07849. URL https://arxiv.org/abs/2310.07849. Version Number: 2.

Qian Liu, Xiaosen Zheng, Niklas Muennighoff, Guangtao Zeng, Longxu Dou, Tianyu Pang, Jing Jiang, and Min Lin. RegMix: Data Mixture as Regression for Language Model Pre-training, January 2025. URL http://arxiv.org/abs/2407.01492. arXiv:2407.01492 [cs].

Zuxin Liu, Thai Hoang, Jianguo Zhang, Ming Zhu, Tian Lan, Shirley Kokane, Juntao Tan, Weiran Yao, Zhiwei Liu, Yihao Feng, Rithesh Murthy, Liangwei Yang, Silvio Savarese, Juan Carlos Niebles, Huan Wang, Shelby Heinecke, and Caiming Xiong. APIGen: Automated Pipeline for Generating Verifiable and Diverse Function-Calling Datasets, June 2024. URL http://arxiv.org/abs/2406.18518. arXiv:2406.18518.

Lin Long, Rui Wang, Ruixuan Xiao, Junbo Zhao, Xiao Ding, Gang Chen, and Haobo Wang. On LLMs-Driven Synthetic Data Generation, Curation, and Evaluation: A Survey, June 2024. URL http://arxiv.org/abs/2406.15126. arXiv:2406.15126 [cs].

Siyu Lu, Jialiang Guo, Shan Liu, Bo Yang, Mingzhe Liu, Lirong Yin, and Wenfeng Zheng. An Improved Algorithm of Drift Compensation for Olfactory Sensors. *Applied Sciences*, 12(19):9529, September 2022. ISSN 2076-3417. doi: 10.3390/app12199529. URL https://www.mdpi.com/2076-3417/12/19/9529.

Ian Magnusson, Nguyen Tai, Ben Bogin, David Heineman, Jena D. Hwang, Luca Soldaini, Akshita Bhagia, Jiacheng Liu, Dirk Groeneveld, Oyvind Tafjord, Noah A. Smith, Pang Wei Koh, and Jesse Dodge. DataDecide: How to Predict Best Pretraining Data with Small Experiments, July 2025. URL http://arxiv.org/abs/2504.11393. arXiv:2504.11393 [cs].

Swaroop Mishra, Daniel Khashabi, Chitta Baral, and Hannaneh Hajishirzi. Cross-Task Generalization via Natural Language Crowdsourcing Instructions. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 3470–3487, Dublin, Ireland, 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.acl-long.244. URL https://aclanthology.org/2022.acl-long.244.

Shikhar Murty, Hao Zhu, Dzmitry Bahdanau, and Christopher D. Manning. NNetNav: Unsupervised Learning of Browser Agents Through Environment Interaction in the Wild, February 2025. URL http://arxiv.org/abs/2410.02907. arXiv:2410.02907 [cs].

Anders Giovanni Møller, Jacob Aarup Dalsgaard, Arianna Pera, and L. Aiello. The Parrot Dilemma: Human-Labeled vs. LLM-augmented Data in Classification Tasks. April 2023. URL https://www.semanticscholar.org/paper/The-Parrot-Dilemma%3A-Human-Labeled-vs.-LLM-augmented-M%C3%B8ller-Dalsgaard/73051b7b25ef972c15ea8e7a221f4361991facbe.

Mihai Nadas, Laura Diosan, and Andreea Tomescu. Synthetic Data Generation Using Large Language Models: Advances in Text and Code. *IEEE Access*, 13:134615–134633, 2025. ISSN 2169-3536. doi: 10.1109/ACCESS.2025.3589503. URL http://arxiv.org/abs/2503.14023. arXiv:2503.14023 [cs].

Yonatan Oren, Nicole Meister, Niladri Chatterji, Faisal Ladhak, and Tatsunori B. Hashimoto. Proving Test Set Contamination in Black Box Language Models. 2023. doi: 10.48550/ARXIV.2310.17623. URL https://arxiv.org/abs/2310.17623. Version Number: 2.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human feedback, March 2022. URL http://arxiv.org/abs/2203.02155. arXiv:2203.02155 [cs].

Karl Pearson and Francis Galton. VII. Note on regression and inheritance in the case of two parents. *Proceedings of the Royal Society of London*, 58(347-352):240–242, January 1997. doi: 10.1098/rspl.1895.0041. URL https://royalsocietypublishing.org/doi/abs/10.1098/rspl.1895.0041.

Krishna Pillutla, Swabha Swayamdipta, Rowan Zellers, John Thickstun, Sean Welleck, Yejin Choi, and Zaid Harchaoui. MAUVE: Measuring the Gap Between Neural Text and Human Text using Divergence Frontiers, November 2021. URL http://arxiv.org/abs/2102.01454. arXiv:2102.01454 [cs].

Joaquin Quinonero-Candela, Masashi Sugiyama, Anton Schwaighofer, and Neil D. Lawrence. *Dataset Shift in Machine Learning*. MIT Press, June 2022. ISBN 978-0-262-54587-7. Google-Books-ID: MBZuEAAAQBAJ.

Qwen, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tianyi Tang, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu. Qwen2.5 Technical Report, January 2025. URL http://arxiv.org/abs/2412.15115. arXiv:2412.15115 [cs].

Sebastian Risi, Sandy D. Vanderbleek, Charles E. Hughes, and Kenneth O. Stanley. How novelty search escapes the deceptive trap of learning to learn. In *Proceedings of the 11th Annual conference on Genetic and evolutionary computation*, GECCO '09, pp. 153–160, New York, NY, USA, July 2009. Association for Computing Machinery. ISBN 978-1-60558-325-9. doi: 10.1145/1569901.1569923. URL https://dl.acm.org/doi/10.1145/1569901.1569923.

Oscar Sainz, Jon Ander Campos, Iker García-Ferrero, Julen Etxaniz, Oier Lopez de Lacalle, and Eneko Agirre. NLP Evaluation in trouble: On the Need to Measure LLM Data Contamination for each Benchmark. arXiv, 2023. doi: 10.48550/ARXIV.2310.18018. URL https://arxiv.org/abs/2310.18018. Version Number: 1.

Erich Schubert and Peter J. Rousseeuw. Faster k-Medoids Clustering: Improving the PAM, CLARA, and CLARANS Algorithms. In Giuseppe Amato, Claudio Gennaro, Vincent Oria, and Miloš Radovanović (eds.), *Similarity Search and Applications*, pp. 171–187, Cham, 2019. Springer International Publishing. ISBN 978-3-030-32047-8. doi: 10.1007/978-3-030-32047-8_16.

Amrith Setlur, Saurabh Garg, Xinyang Geng, Naman Garg, Virginia Smith, and Aviral Kumar. RL on Incorrect Synthetic Data Scales the Efficiency of LLM Math Reasoning by Eight-Fold. 2024. doi: 10.48550/ARXIV.2406.14532. URL https://arxiv.org/abs/2406.14532. Version Number: 1.

Ilia Shumailov, Zakhar Shumaylov, Yiren Zhao, Nicolas Papernot, Ross Anderson, and Yarin Gal. AI models collapse when trained on recursively generated data. *Nature*, 631(8022):755–759, July 2024. ISSN 0028-0836, 1476-4687. doi: 10.1038/s41586-024-07566-y. URL https://www.nature.com/articles/s41586-024-07566-y.

C. Spearman. The Proof and Measurement of Association between Two Things. *The American Journal of Psychology*, 15(1):72, January 1904. ISSN 00029556. doi: 10.2307/1412159. URL https://www.jstor.org/stable/1412159?origin=crossref.

Qiushi Sun, Kanzhi Cheng, Zichen Ding, Chuanyang Jin, Yian Wang, Fangzhi Xu, Zhenyu Wu, Chengyou Jia, Liheng Chen, Zhoumianze Liu, Ben Kao, Guohao Li, Junxian He, Yu Qiao, and Zhiyong Wu. OS-Genesis: Automating GUI Agent Trajectory Construction via Reverse Task Synthesis, December 2024. URL http://arxiv.org/abs/2412.19723. arXiv:2412.19723 [cs].

Zhen Tan, Dawei Li, Song Wang, Alimohammad Beigi, Bohan Jiang, Amrita Bhattacharjee, Mansooreh Karami, Jundong Li, Lu Cheng, and Huan Liu. Large Language Models for Data Annotation and Synthesis: A Survey, December 2024. URL http://arxiv.org/abs/2402.13446. arXiv:2402.13446 [cs].

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. LLaMA: Open and Efficient Foundation Language Models, February 2023. URL http://arxiv.org/abs/2302.13971. arXiv:2302.13971 [cs].

Tianbao Xie, Danyang Zhang, Jixuan Chen, Xiaochuan Li, Siheng Zhao, Ruisheng Cao, Toh Jing Hua, Zhoujun Cheng, Dongchan Shin, Fangyu Lei, Yitao Liu, Yiheng Xu, Shuyan Zhou, Silvio Savarese, Caiming Xiong, Victor Zhong, and Tao Yu. OSWorld: Benchmarking Multimodal Agents for Open-Ended Tasks in Real Computer Environments, May 2024. URL http://arxiv.org/abs/2404.07972. arXiv:2404.07972 [cs].

Yiheng Xu, Dunjie Lu, Zhennan Shen, Junli Wang, Zekun Wang, Yuchen Mao, Caiming Xiong, and Tao Yu. AgentTrek: Agent Trajectory Synthesis via Guiding Replay with Web Tutorials, December 2024. URL http://arxiv.org/abs/2412.09605. arXiv:2412.09605 [cs].

Jiaxi Yang, Binyuan Hui, Min Yang, Jian Yang, Junyang Lin, and Chang Zhou. Synthesizing Text-to-SQL Data from Weak and Strong LLMs, August 2024. URL http://arxiv.org/abs/2408.03256. arXiv:2408.03256 [cs].

Jiacheng Ye, Jiahui Gao, Qintong Li, Hang Xu, Jiangtao Feng, Zhiyong Wu, Tao Yu, and Lingpeng Kong. ZeroGen: Efficient Zero-shot Learning via Dataset Generation. In Yoav Goldberg, Zornitsa Kozareva, and Yue Zhang (eds.), *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pp. 11653–11669, Abu Dhabi, United Arab Emirates, December 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.emnlp-main.801. URL https://aclanthology.org/2022.emnlp-main.801.

Yue Yu, Yuchen Zhuang, Jieyu Zhang, Yu Meng, Alexander Ratner, Ranjay Krishna, Jiaming Shen, and Chao Zhang. Large Language Model as Attributed Training Data Generator: A Tale of Diversity and Bias. 2023. doi: 10.48550/ARXIV.2306.15895. URL https://arxiv.org/abs/2306.15895. Version Number: 2.

Chi Zhang, Huaping Zhong, Kuan Zhang, Chengliang Chai, Rui Wang, Xinlin Zhuang, Tianyi Bai, Jiantao Qiu, Lei Cao, Ju Fan, Ye Yuan, Guoren Wang, and Conghui He. Harnessing Diversity for Important Data Selection in Pretraining Large Language Models, October 2024. URL http://arxiv.org/abs/2409.16986. arXiv:2409.16986 [cs].

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. Judging LLM-as-a-Judge with MT-Bench and Chatbot Arena, December 2023. URL http://arxiv.org/abs/2306.05685. arXiv:2306.05685 [cs].

Yifei Zhou, Qianlan Yang, Kaixiang Lin, Min Bai, Xiong Zhou, Yu-Xiong Wang, Sergey Levine, and Erran Li. Proposer-Agent-Evaluator(PAE): Autonomous Skill Discovery For Foundation Model Internet Agents, December 2024. URL http://arxiv.org/abs/2412.13194. arXiv:2412.13194 [cs].

Yuchang Zhu, Huizhe Zhang, Bingzhe Wu, Jintang Li, Zibin Zheng, Peilin Zhao, Liang Chen, and Yatao Bian. Measuring Diversity in Synthetic Datasets, February 2025. URL http://arxiv.org/abs/2502.08512. arXiv:2502.08512 [cs].

A Visualization of ambiguous image

(a) Sample synthetic images of class "stage"

(b) Sample synthetic images of class "throne"

Figure 2: Visualization of synthetic images from second split for classes (a) "stage" and (b) "throne"

B Additional Ablations

B.1 Ablation Study on Lens Preferential Bias

Table 7: We use two non-Qwen scoring models on tasks where synthetic data was generated by `Qwen2.5` models. The consistent positive correlations of the debiased scores demonstrate that LENS is robust to this potential bias.

LENS (`Granite-8B`)	Debiased		Biased	
Task	Spearman	Pearson	Spearman	Pearson
Sentiment	.21	.30	-.04	-.02
Text2SQL	.22	.20	.19	.23
LENS (`Ministral-8B`)	Debiased		Biased	
Task	Spearman	Pearson	Spearman	Pearson
Sentiment	.28	.39	.52	.31
Text2SQL	.32	.29	.16	.16

B.2 Lens with Different Backbone Models and Sizes

As shown in table 8, LENS demonstrates robust ranking capabilities across a diverse range of backbone models and sizes. However, the impact of the debiasing step varies significantly by task complexity and model capability:

- **Text2SQL (Complex Generation):** Debiasing consistently improves or maintains correlation strength across nearly all models. For instance, with `Ministral-8B`, debiasing doubles the Spearman correlation (.16 $\rightarrow$ .32), and it yields the highest overall performance with `Qwen2.5-32B` (.46). This suggests that for complex structural tasks, removing distributional priors remains beneficial even for capable models.

- **Sentiment Analysis (Simple Classification):** The results are more nuanced. For models with severe initial bias, such as `Granite-8B-Instruct`, debiasing is critical, correcting a negative correlation (-.04) to a positive one (.21). However, for highly capable or larger models (e.g., `GPT-4.1`, `Granite-4.0 (32B)`, `Gemma3-27B`), debiasing tends to degrade performance. We hypothesize that these stronger models are already well-calibrated for simple classification tasks; consequently, our analytical debiasing term may introduce unnecessary noise that overrides the model's accurate intrinsic priors.

Table 8: Comparison of original and debiased Lens scores for ranking synthetic datasets across various backbone models and sizes. For GPT-4.1, sentiment analysis results are reported only due to cost constraints.

LENS (Qwen2.5-7B-Instruct)	Debiased		Original	
Task	Spearman	Pearson	Spearman	Pearson
Sentiment	.25	.33	.26	.17
Text2SQL	.33	.33	.31	.28
LENS (Qwen2.5-32B-Instruct)	Debiased		Original	
Task	Spearman	Pearson	Spearman	Pearson
Sentiment	.38	.26	.24	.23
Text2SQL	.46	.43	.43	.44
LENS (Ministral-8B-Instruct)	Debiased		Original	
Task	Spearman	Pearson	Spearman	Pearson
Sentiment	.28	.39	.52	.31
Text2SQL	.32	.29	.16	.16
LENS (Granite-8B-Instruct)	Debiased		Original	
Task	Spearman	Pearson	Spearman	Pearson
Sentiment	.21	.30	-.04	-.02
Text2SQL	.22	.20	.19	.23
LENS (Granite-4.0-h-small) (32B)	Debiased		Original	
Task	Spearman	Pearson	Spearman	Pearson
Sentiment	.16	.39	.50	.44
Text2SQL	.32	.36	.40	.48
LENS (Gemma3-12B-Instruct)	Debiased		Original	
Task	Spearman	Pearson	Spearman	Pearson
Sentiment	.10	.13	.30	.18
Text2SQL	.30	.34	.25	.25
LENS (Gemma3-27B-Instruct)	Debiased		Original	
Task	Spearman	Pearson	Spearman	Pearson
Sentiment	.17	.33	.37	.31
Text2SQL	.31	.38	.31	.27
LENS (GPT-4.1-2025-04-14)	Debiased		Original	
Task	Spearman	Pearson	Spearman	Pearson
Sentiment	.07	.30	.28	.37

B.3 Ablation Study on Lens Rubric Design

Our ablation results from table 9 demonstrate the effectiveness of the rubric component of Lens. Table 10 demonstrates that generating too few or too many points will degrade the performance of Lens.

Table 9: LENS with vs. without a rubric on sentiment analysis, showing the rubric's effectiveness. We use `Qwen2.5-7B-Instruct` for scoring.

LENS Model	**w/ Rubric**	**Spearman**	**Pearson**
Qwen2.5-7B-Instruct	NO	0.23	0.21
Qwen2.5-32B-Instruct	NO	0.32	0.13
Qwen2.5-7B-Instruct	YES	0.25	0.33
Qwen2.5-32B-Instruct	YES	0.38	0.26

Table 10: Bottom: Varying the number of rubric points, where 10 points provides a good balance.

# Rubric Points	**Sentiment Analysis**		**Text2SQL (Avg)**	
(Debiased)	Spearman	Pearson	Spearman	Pearson
5	0.14	0.19	0.38	0.30
10	0.25	0.33	0.33	0.33
15	-0.06	0.05	0.23	0.21

B.4 Ablation Study on MMD^2 Kernel Functions

Table 11: Pearson / Spearman correlation coefficients of $\mathbf{MMD}^2$ with different kernel functions.

Task	**Polynomial**	**RBF**	**Laplacian**	**Linear**	**Sigmoid**
Sentiment	.67 / .45	.67 / .45	**.78** / **.60**	.67 / .45	.67 / .46
Text2SQL (avg)	.51 / .43	.52 / .44	**.70** / **.51**	.52 / .44	.53 / .44
Image (avg)	.22 / .30	.22 / .30	.21 / .30	.22 / .30	.22 / .30
WebNav	.06 / -.02	.06 / -.01	.05 / -.04	.06 / -.02	.05 / -.03

Based on the ablation results from table 11, the Laplacian kernel performs best on text tasks, but all kernels show limited effectiveness on more complex domains.

B.5 Ablation Study on MDM with Different Number of Clusters

Table 12: Pearson / Spearman coefficients of **MDM** with different number of clusters (K=3, 5, 10, 20) on sentiment analysis.

K=3	**K=5**	**K=10**	**K=20**
.85/ .68	.86/ .68	.86/ .69	.86/ .69

The table with varying number of medoid clusters for **MDM** shows that changing the number of clusters has negligible effect on the correlation between **MDM** and F1 score.

B.6 Ablation Study on Using Different Encoder for Representation-Based Metrics

The results with BGE-M3[6] in table 13 demonstrate that representation-based metrics are measuring fundamental distribution shifts rather than artifacts of a specific encoder, therefore indicating the effectiveness of representation-based metrics.

Table 13: Pearson / Spearman correlation coefficients of proxies on sentiment analysis with BGE-M3 as the encoder.

Mauve	**PAD**	**MMD^2**	**MDM**
.60/ .65	.75/ .65	.82/ .67	.86/ .74

B.7 Correlation of PAD Predictions from Different Classifiers vs. F1 Scores

Table 14: Correlation Coefficients (Pearson / Spearman) of **PAD** predictions from different classifiers vs. F1 scores.

Classifier	**Correlation Coefficients (P/S)**
Logistic Regression	.65 / .53
Random Forest	.75 / .64
2-layer MLP	.56 / .43

In summary, according to table 14, classifier choice plays a critical role in **PAD** prediction performance, with the random forest model demonstrating the highest correlation with F1 scores. This may be due to the ensemble nature of both the random forest and the sentiment analysis classifier (XGBoost), possibly enhancing alignment between predictions and F1 outcomes, while the MLP shows weaker correspondence.

B.8 Trade-Off Analysis Between Lens and Diversity Measure

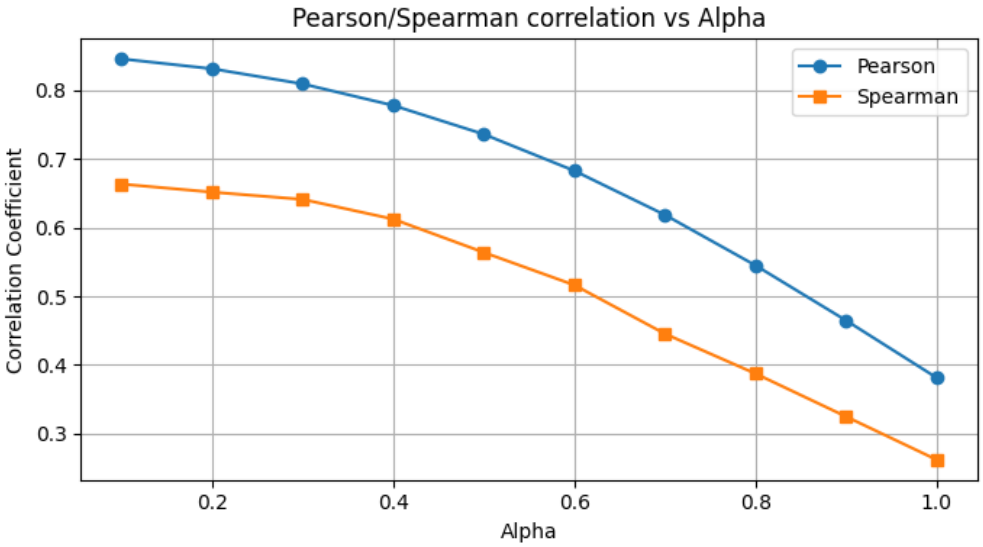


Figure 3: Pearson/Spearman correlation coefficients of the hybrid score vs. alpha. Both correlation coefficients are monotonically decreasing as alpha increases (more LENS impact and less **MDM** impact).

[6] https://huggingface.co/BAAI/bge-m3

To further understand the interplay between LENS and the diversity measure, we conducted an experiment using a "hybrid" score, defined as $\alpha \cdot \text{LENS} + (1 - \alpha) \cdot \textbf{MDM}$. As shown in fig. 3, our analysis reveals that increasing α – thereby placing greater weight on LENS and less on diversity – results in a monotonic decrease in both the Pearson and Spearman correlation coefficients. This trend suggests that the LENS and diversity measures do not demonstrate synergistic effects when combined.

B.9 Analysis on Using Hybrid Proxies for Dataset Selection

Table 15: Top 10 hybrid proxies by Spearman (ρ) and Pearson (r) correlation for dataset selection in the Sentiment Analysis (upper block) and Text2SQL (lower block) domains. Each row details a specific weighted combination of **PAD**, **MMD^2**, **MDM**, LENS, and MAUVE, alongside its resulting correlation to downstream task performance. For baseline comparison, $\hat{\rho}$ and $\hat{r}$ denote the maximum correlation achieved by the best-performing individual metric from table 3, whereas ρ and r denote the correlation achieved by the hybrid combination.

Spearman							**Pearson**						
$\hat{\rho}$	ρ	**PAD**	**MMD^2**	**MDM**	**Lens**	**Mauve**	$\hat{r}$	r	**PAD**	**MMD^2**	**MDM**	**Lens**	**Mauve**
Sentiment Analysis													
0.68	0.72	0.0	0.4	0.8	0.0	0.2	0.85	0.85	0.0	0.0	0.2	0.0	0.0
0.68	0.72	0.0	0.6	1.0	0.0	0.4	0.85	0.85	0.0	0.0	0.4	0.0	0.0
0.68	0.72	0.0	0.4	0.6	0.0	0.2	0.85	0.85	0.0	0.0	0.6	0.0	0.0
0.68	0.71	0.0	0.6	0.8	0.0	0.4	0.85	0.85	0.0	0.0	0.8	0.0	0.0
0.68	0.71	0.2	0.6	1.0	0.0	0.6	0.85	0.85	0.0	0.0	1.0	0.0	0.0
0.68	0.71	0.0	0.6	1.0	0.0	0.2	0.85	0.85	0.2	0.0	1.0	0.0	0.2
0.68	0.71	0.0	0.8	1.0	0.0	0.6	0.85	0.85	0.2	0.0	0.8	0.0	0.2
0.68	0.71	0.2	0.4	0.8	0.0	0.4	0.85	0.85	0.0	0.0	1.0	0.0	0.2
0.68	0.71	0.0	0.8	1.0	0.0	0.4	0.85	0.85	0.0	0.2	1.0	0.0	0.2
0.68	0.71	0.0	0.2	1.0	0.0	0.0	0.85	0.84	0.0	0.0	0.8	0.0	0.2
Text2SQL													
0.65	0.89	0.6	0.8	0.4	0.6	0.2	0.68	0.79	0.0	1.0	0.2	0.8	0.2
0.65	0.89	0.6	0.6	0.4	0.6	0.2	0.68	0.79	0.0	0.8	0.2	0.8	0.2
0.65	0.89	0.2	0.2	0.6	0.8	0.2	0.68	0.79	0.0	0.6	0.2	0.8	0.2
0.65	0.89	0.0	0.2	0.6	0.8	0.2	0.68	0.79	0.0	0.4	0.2	0.8	0.2
0.65	0.89	0.2	0.4	0.4	0.6	0.2	0.68	0.79	0.0	0.2	0.2	0.8	0.2
0.65	0.89	0.4	1.0	0.4	0.6	0.2	0.68	0.79	0.0	0.0	0.2	0.8	0.2
0.65	0.89	0.2	0.2	0.4	0.6	0.2	0.68	0.79	0.0	1.0	0.2	1.0	0.2
0.65	0.89	0.4	0.6	0.4	0.6	0.2	0.68	0.79	0.0	0.8	0.2	1.0	0.2
0.65	0.89	0.4	0.4	0.4	0.6	0.2	0.68	0.79	0.0	0.6	0.2	1.0	0.2
0.65	0.89	0.6	0.2	0.4	0.6	0.2	0.68	0.79	0.0	0.4	0.2	1.0	0.2

To understand the synergistic potential of different synthetic data metrics, we evaluated over 7,700 linear combinations (6 settings for each proxy, ranging from 0.0 to 1.0 with a step size of 0.2, resulting in 6^5 combinations) of our five primary proxies. We expanded this experiment to include both the Sentiment Analysis and Text2SQL "movie_platform" domains (as this domain provides the most instances and is therefore a more stable Text2SQL domain) to analyze how hybrid performance scales with task complexity. Here are the core findings based on results from table 15:

- **Accurate Dataset Ranking Requires Synergy:** While individual metrics establish a baseline, combining them yields strict improvements, particularly for rank-ordering. In Sentiment Analysis,

blending metrics increases the maximum single-proxy Spearman correlation ($\max \rho$) from 0.68 to 0.72. In the more complex Text2SQL domain, the synergistic gains are significant: the hybrid approach boosts the maximum single-proxy Spearman correlation ($\max \rho$) from 0.65 to 0.89, and the maximum Pearson correlation ($\max r$) from 0.68 to 0.79.

- **The Optimal Blend Varies from Task to Task:** The optimal hybrid composition shifts significantly depending on the nature of the task – there is no single combination of proxies that performs the best across tasks. For simple stylistic tasks like Sentiment Analysis, representation-based metrics dominate the top combinations, with **MDM** acting as a primary anchor. However, for structurally complex reasoning tasks like Text2SQL, LENS becomes a critical component for both Pearson and Spearman correlations, consistently requiring heavy weights (between 0.6 and 1.0) to achieve optimal utility prediction.
- **Proxies Capture Orthogonal Blind Spots:** The necessity of this blend confirms that these metrics measure fundamentally distinct properties. A synthetic dataset might exhibit excellent diversity but drift out-of-domain, or it might perfectly mimic the real distribution's style while suffering from severe mode collapse. Furthermore, representation metrics may miss nuanced logical errors in complex queries that LLM-based reasoning (LENS) catches. The top-performing hybrid proxies act as a counterbalance, successfully penalizing datasets that fail on any front.

Overall, if the goal is to select the absolute best synthetic dataset for downstream training, practitioners must use a holistic scoring approach that demands broad embedding coverage (i.e., diversity), strict distributional alignment, and – particularly for complex tasks – explicit reasoning-based evaluation.

B.10 Ablation on the Number of Samples Used for Scoring on Simple Task

Table 16: Pearson / Spearman correlation between the LENS score and the test F1 score for different metrics with differet number of real instances used for scoring (100 and 200).

$\lvert\mathcal{U}_\mathrm{r}\rvert = 100$					$\lvert\mathcal{U}_\mathrm{r}\rvert = 200$				
LENS	**PAD**	**MMD^2**	**MDM**	MAUVE	LENS	**PAD**	**MMD^2**	**MDM**	MAUVE
-.03/-.13	.65/.47	.68/.44	.85/.70	.54/.51	.26/.38	.65/.53	.67/.45	.85/.68	.57/.53

We investigate the sensitivity of LENS proxies to the size of the unannotated real dataset $|\mathcal{U}_r|$ for the sentiment analysis task.As shown in table 16, representation-based metrics (**PAD**, **MMD^2**, **MDM**, MAUVE) demonstrate remarkable robustness in this domain, maintaining strong correlations (e.g., **MDM** Spearman 0.70) even with as few as 100 samples. This aligns with our hypothesis that embedding spaces effectively capture distributional shifts in simpler, stylistic tasks. In contrast, LENS requires a larger sample size to stabilize for this specific task 5, exhibiting weak correlations at $|\mathcal{U}_r| = 100$ (Pearson -0.03, Spearman -0.13) before recovering significantly to a positive Spearman correlation of 0.38 at $|\mathcal{U}_r| = 200$. This behavior highlights a distinct complexity trade-off: while representation metrics are highly data-efficient for stylistic domains, they fail on complex, long-horizon tasks such as Web Navigation. Conversely, LENS proves robust in those more complex domains even with very limited unlabled real data.

B.11 Correlation Improvement with De-Hallucination Prompt

Table 17: Spearman and Pearson correlations for baseline and with de-hallucination prompt (w/ DHP), across different models.

	Baseline	w/ DHP (Delta)
Qwen2.5-7B-Instruct		
Spearman	0.25	0.15 (−0.10)
Pearson	0.33	0.30 (−0.03)
Gemma-3-12B-It		
Spearman	0.10	0.19 (+0.09)
Pearson	0.13	0.24 (+0.11)
Ministral-8B-Instruct-2410		
Spearman	0.28	0.53 (+0.25)
Pearson	0.39	0.49 (+0.10)
Granite3.3-8B-Instruct		
Spearman	0.21	0.23 (+0.02)
Pearson	0.30	0.32 (+0.02)

As shown in Table 17, incorporating an explicit de-hallucination prompt during the LLM-based scoring phase produces highly model-dependent effects – four out of three models improve correlations.

To incorporate de-hallucination into the LENS scoring prompt, we add the following sentence:

```
If you find the headline is hallucinated (i.e., not a real headline), answer "unsure".
```

We find that the impact of de-hallucination prompts on correlation metrics is highly model-dependent: while some models benefit noticeably with improved alignment (e.g., substantial gains in proxy reliability for certain models), others show little change or even reduced correlations. This indicates that prompt-based de-hallucination provides inconsistent results across models, suggesting that more robust, systematic approaches are needed to reliably reduce hallucinations during evaluation.

B.12 Ablation on the Size of Synthetic Dataset for SynQuE Proxies

Table 18: Scaling analysis of SynQuE proxies for sentiment analysis. Table entries show Pearson/Spearman correlation coefficients, in the form "Pearson/Spearman", for each proxy at different synthetic dataset size multiples. We report results using debiased Lens, evaluated with `Qwen2.5-(7B/32B)-Instruct` scorers. A size multiple of n indicates that n synthetic datasets (each containing 999 samples) are merged, yielding $n \times 999$ total samples.

Size	LENS-7B	LENS-32B	MDM	MMD^2	PAD	Mauve
1	.33/.25	.26/.38	.85/.68	.67/.45	.65/.53	.57/.53
2	.17/.21	.11/.16	.69/.52	.55/.50	.51/.43	.35/.36
4	.17/.12	.38/.30	-.70/-.81	-.27/-.42	-.25/-.43	-.31/-.32
8	.04/.04	.49/.44	-.51/-.20	.62/.40	.04/.40	-.20/.24

We perform a scaling experiment on sentiment analysis to evaluate how well SynQuE proxies maintain their effectiveness as the size of synthetic datasets increases. Sentiment analysis provides the most abundant synthetic training data (32 datasets), enabling us to systematically merge n datasets together, $n \in \{1, 2, 4, 8\}$.

From the results in table 18, we observe that increasing the synthetic dataset size generally leads to performance degradation across all proxies. In particular, when the size multiple reaches 4, LENS remains capable of achieving a positive correlation, suggesting greater robustness to growing dataset size compared to other methods. It is important to note that, throughout these experiments, the number of synthetic samples chosen for rubric compilation in LENS is fixed at 200, which means that as dataset size increases, the relative proportion of data represented in the scoring rubric decreases, potentially limiting its ability to capture the full variance present in larger datasets.

B.13 Ablation on Instance-Level Data Selection

Table 19: Instance-level downstream task performance (%) on sentiment analysis for synthetic datasets selected by each SYNQUE proxy. Debiased LENS uses `Qwen2.5-7B-Instruct` as the backbone model for scoring

Proxy	**Average**	LENS	**PAD**	**MMD^2**	**MDM**	MAUVE
Result	49.6	54.8	55.9	53.2	n/a	52.7

We explore whether SYNQUE proxies can be leveraged for *instance-level* data selection, as opposed to the default *dataset-level* ranking. In this ablation, for sentiment analysis, we aggregate all deduplicated synthetic datasets into a single pool containing 29,235 instances. From this pool, for each class, we select the top 333 instances according to each proxy score, yielding a final synthetic dataset of 999 samples (333 per class).

The results in table 19 reveal: *discriminative* proxies – namely **PAD** and LENS – achieve higher downstream task performance when selecting at the instance level, compared to their performance with dataset-level selection. Specifically, incorporating comparisons to table 2, instance-level selection improves **PAD**'s performance (55.9% vs. 55.3%) and LENS's performance (54.8% vs. 50.5%) over the dataset-level setting.

In contrast, for the other proxies such as **MMD^2** and MAUVE, switching from dataset-level to instance-level selection leads to reduced task performance, indicating that these methods are less suitable for fine-grained instance selection. We also note that **MDM** is not applicable at the instance level, since it measures diversity across a set of samples and cannot score individual ones – hence it is marked as "n/a" in the table.

Overall, these results demonstrate that discriminative proxies can better exploit instance-level selection, whereas distributional measures suffer from a loss of signal in this setting.

C Additional Experimental Details

C.1 Data Curation

Sentiment analysis For sentiment analysis, we used a cleaned version of the original validation split as the test set, removing URLs from the data. Synthetic samples are generated and validated to ensure each output is non-empty, alphabetic 'headline' and a 'sentiment' label restricted to the values '0', '1', or '2'. Only samples meeting these structural and content criteria were retained for downstream analysis.

Text2SQL During Text2SQL data synthesis, we validate generated question–SQL pairs by ensuring both fields are non-empty, contain alphanumeric content, and are formatted as a dictionary with 'question' and 'SQL' keys. For non-zero-shot generations, SQL queries are executed against the target database; pairs failing execution are discarded. This process enforces correct structure, meaningful content, and SQL executability.

Image classification We classify each image in the `unmet-promise` dataset into one of three categories: *label_only*, *label_relation*, or *label_background*, based on its caption. Captions are lowercased and stemmed. If a class-specific background keyword appears in the caption, the image is assigned to *label_background*. If a relation keyword is present, it is assigned to *label_relation*. Images not matching either are assigned to *label_only*. We balance the number of samples per category and class, and store the processed data for downstream tasks. To further ensure fidelity of images, we use `Qwen2.5-VL-7B-Instruct` to filter noisy

images. Prompt used for filtering is provided in listing 8. For image classification, we create a test set for each split by selecting 150 samples that are balanced across all labels.

Web navigation We first group individual trajectories into tasks. Then we use different seeds to create 5 disjoint subsets with equal amount of tasks. We fine-tune using LoRA rank of 64 and the open-instruct [7] fine-tuning codebase.

C.2 SynQuE Scoring

For **PAD**, we reserve 20% of all embeddings as a holdout test set to train the classifier, and compute the classification error on this set to obtain the final **PAD** score. For $\mathbf{MMD}^2$, we generate the proxy score using a polynomial kernel [8] with degree 3 and Coef0 parameter 1. For **MDM**, we use Fasterpam (Schubert & Rousseeuw, 2019) to compute k medoids in the embeddings, setting k to 3 for sentiment analysis and 5 for other tasks, using Euclidean distance for clustering; **MDM** is then calculated by averaging the Euclidean distance from each data point to its corresponding medoid. For Lens, the prompts used for rubric compilation and scoring are provided in section C.8.3 and section C.8.4, respectively. For Mauve calculation, we use the default hyperparameter, the same embedding model as the other representation-based proxies (*qte-Qwen2-7B-Instruct*), and the scaling factor is 5. For Perplexity calculation in Text2SQL, we first reformat each Text2SQL question as a string: `"question": <sample>`. We then use `Qwen2.5-7B` to compute the perplexity, considering only the tokens corresponding to the question text (i.e., the <sample> portion), and excluding the prompt tokens (such as `"question":`).

[7] https://github.com/allenai/open-instruct

[8] https://scikit-learn.org/stable/modules/generated/sklearn.metrics.pairwise.polynomial_kernel.html

C.3 Dataset Size Descriptions

Table 20: Dataset sizes for synthetic ($\mathcal{D}_{\text{s}}$), rubric selection, real subset ($\mathcal{U}_{\text{r}}$), and full real dataset ($\mathcal{D}_{\text{r}}$) for all tasks.

Task	Synthetic	LENS Rubric Compilation	Real Subset	Full Real
Sentiment Analysis	999	400	200	2388
		Text-to-SQL		
Movie Platform	1000	60	30	164
App Store	1000	60	30	60
Computer Student	1000	60	30	69
Image Classification	1500	200	100	150
		Web Navigation		
Allrecipes	79	99	20	45
Amazon	63	83	20	41
Apple	70	90	20	43
ArXiv	80	100	20	43
BBC	69	89	20	42
Coursera	72	92	20	42
dictionary	54	74	20	43
ESPN	62	82	20	44
GitHub	71	91	20	41
Google Maps	75	95	20	41
Google Search	72	92	20	43
Hugging Face	76	96	20	43
Wolfram Alpha	66	86	20	46

Table 21: Number of synthetic datasets available per task.

Task	Number of Synthetic Datasets
Sentiment Analysis	32
Text-to-SQL	8
Image Classification	6
Web Navigation	65

C.4 Standard deviation of correlation coefficients

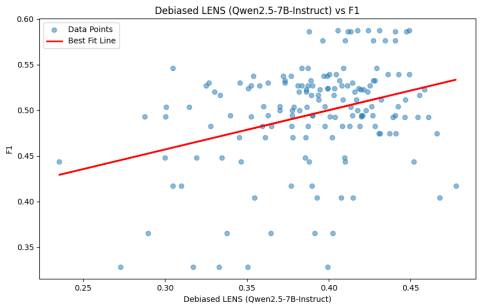
Table 22: Standard deviation of Spearman (left) and Pearson (right) correlation coefficients of SynQuE proxy metrics across 5 seeds. **MDM** only scores on synthetic datasets therefore no change in input data.

Tasks	Lens 7B		Lens 32B		PAD	MMD	MDM
	debiased	biased	debiased	biased			
Sentiment	.23 .17	.12 .14	.09 .11	.07 .07	.02 .02	.02 .02	.00 .00
Text2SQL							
Computer	.48 .39	.38 .44	.21 .21	.18 .14	.04 .08	.03 .08	.00 .00
Apps	.38 .34	.25 .21	.22 .22	.22 .14	.36 .48	.05 .15	.00 .00
Movies	.32 .32	.33 .24	.22 .24	.11 .14	.07 .13	.05 .14	.00 .00
Average	.39 .35	.32 .30	.22 .22	.17 .14	.15 .23	.04 .12	.00 .00
Image							
Split 1	.47 .48	.51 .51	.39 .38	.28 .22	.08 .06	.00 .02	.00 .00
Split 2	.43 .47	.43 .42	.40 .41	.43 .53	.11 .08	.00 .04	.00 .00
Split 3	.52 .39	.47 .39	.40 .42	.29 .22	.07 .07	.00 .05	.00 .00
Average	.47 .44	.47 .44	.40 .40	.33 .32	.09 .07	.00 .04	.00 .00
WebNav	.15 .17	.11 .13	.09 .11	.15 .19	.04 .02	.02 .02	.00 .00

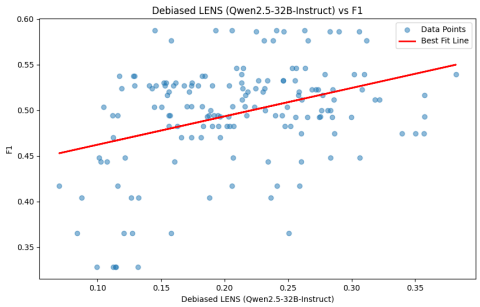
C.5 Scatter Plot of Lens Scores and Task Utility

C.5.1 Sentiment Analysis

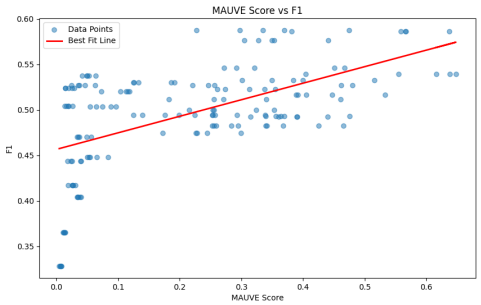
Scatter plots of debiased SynQuE scores and baselines vs. F1 utility on sentiment analysis datasets. Note that we use the raw scores of $\mathbf{MMD}^2$ and **PAD** – we negate them to have a positive correlation with task utility for dataset selection.



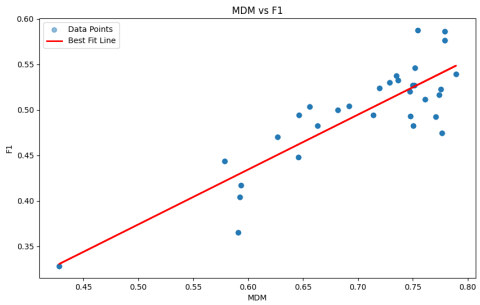
(a) Debiased LENS (`Qwen2.5-7B-Instruct`) vs. F1



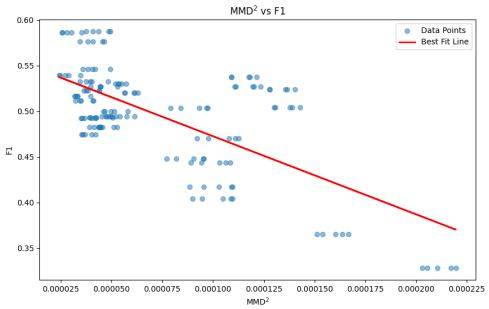
(b) Debiased LENS (`Qwen2.5-32B-Instruct`) vs. F1



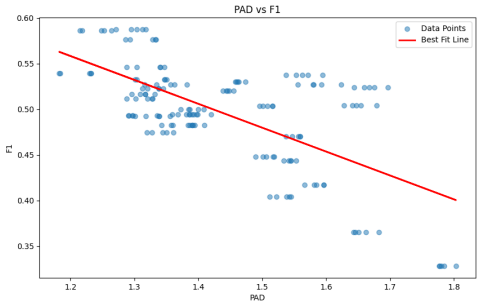
(c) MAUVE vs. F1



(d) **MDM** vs. F1



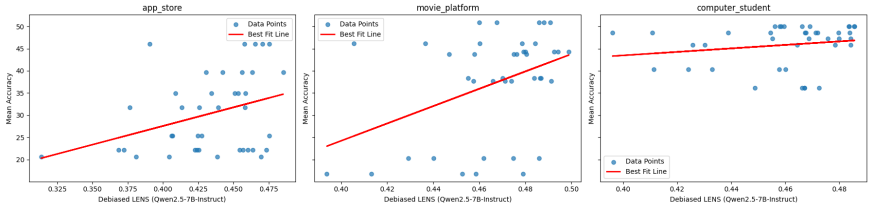
(e) **MMD**^{2} vs. F1



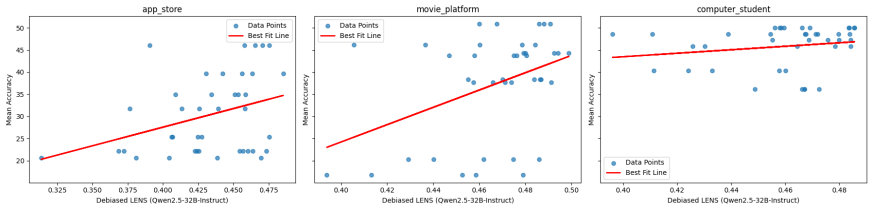
(f) **PAD** vs. F1

C.5.2 Text-to-SQL

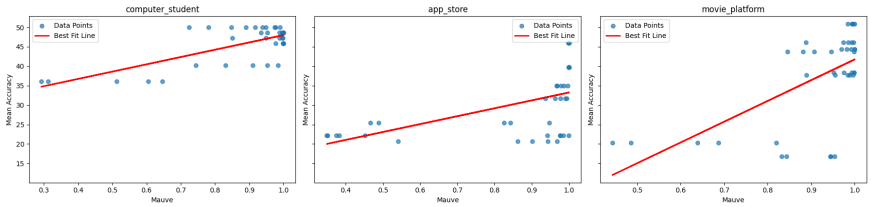
Below are scatter plots of SynQuE scores vs. accuracy utility on text-to-SQL datasets. For **MMD**^{2} and **PAD**, we use raw scores and negate them for positive correlation with task utility for dataset selection.



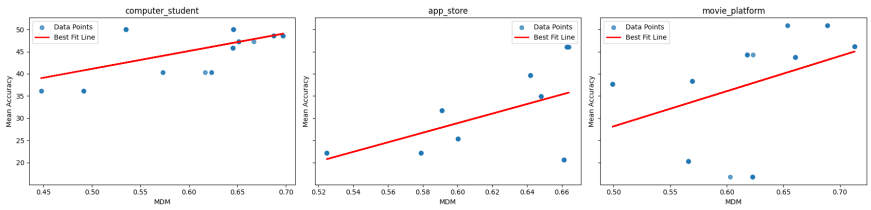
(a) Debiased Lens (`Qwen2.5-7B-Instruct`) vs. Accuracy



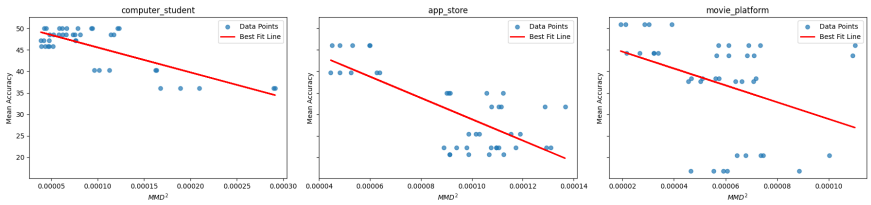
(b) Debiased Lens (`Qwen2.5-32B-Instruct`) vs. Accuracy



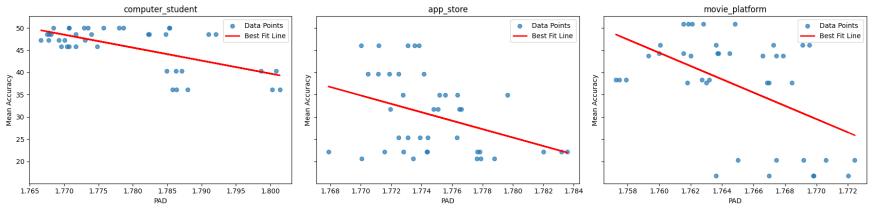
(c) Mauve vs. Accuracy



(d) **MDM** vs. Accuracy



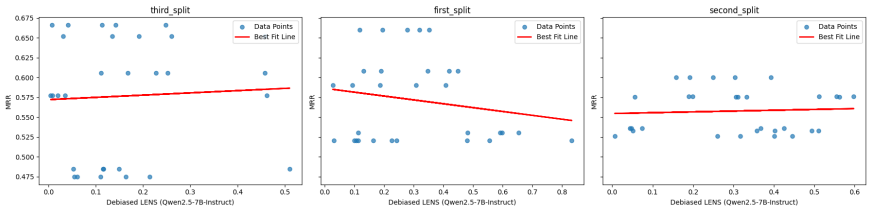
(e) **MMD**^{2} vs. Accuracy



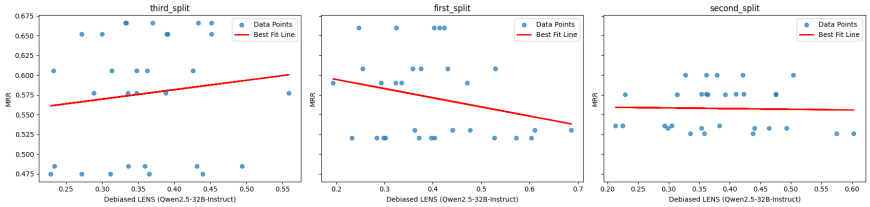
(f) **PAD** vs. Accuracy

C.5.3 Image Classification

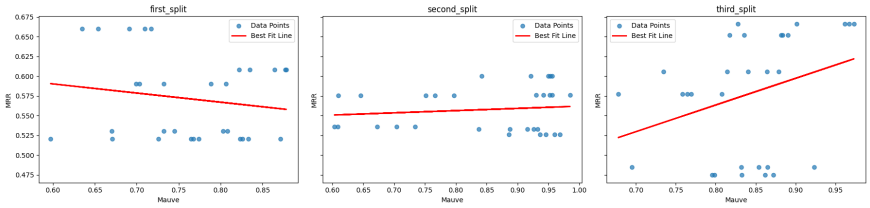
Below are scatter plots SynQuE scores vs. MRR utility on image classification datasets. For **MMD**^{2} and **PAD**, we use raw scores and negate them for positive correlation with task utility for dataset selection.



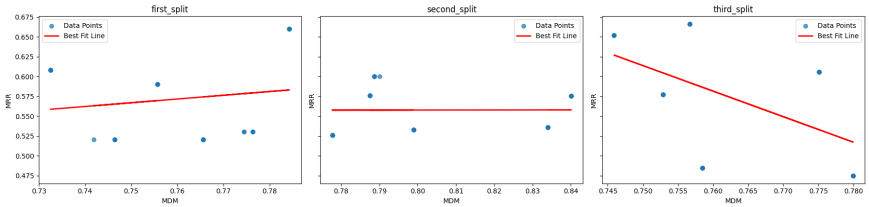
(a) Debiased Lens (`Qwen2.5-7B-Instruct`) vs. MRR



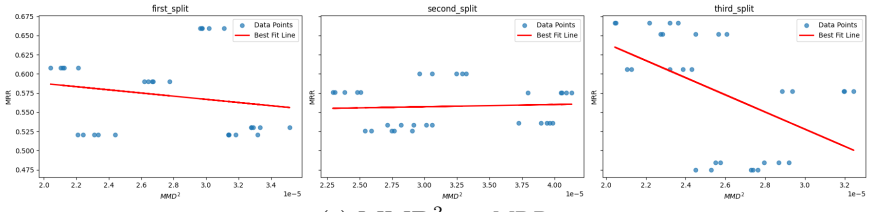
(b) Debiased Lens (`Qwen2.5-32B-Instruct`) vs. MRR



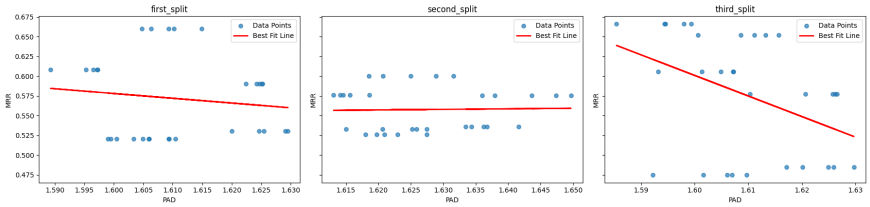
(c) Mauve vs. MRR



(d) **MDM** vs. MRR



(e) **MMD**^{2} vs. MRR



(f) **PAD** vs. MRR

C.6 SynQuE proxy cost and efficiency analysis

Table 23: Comparison of computational cost across different proxy methods in sentiment analysis. Columns: **Emb.** (embedding latency), **Score** (scoring latency), **Total** (end-to-end latency), and **Tokens** (total tokens for Lens, including input and output tokens). Latencies reported in seconds per synthetic dataset (999 samples), with standard deviation in parentheses. Lens uses Qwen2.5 models.

Method	Emb. (s/$\mathcal{D}$)	Score (s/$\mathcal{D}$)	Total (s/$\mathcal{D}$)	Num. of Tokens
Lens(7B)	–	90.52 ± 4.84	90.52 ± 4.84	9264 ± 440
Lens(32B)	–	271.25 ± 20.54	271.25 ± 20.54	9264 ± 440
PAD	6.08 ± 0.66	1.99 ± 0.29	8.07 ± 0.72	–
MDM	6.08 ± 0.66	0.24 ± 0.01	6.32 ± 0.66	–
MMD^2	6.08 ± 0.66	0.04 ± 0.01	6.12 ± 0.66	–
Mauve	6.08 ± 0.66	106.43 ± 6.30	112.51 ± 6.33	–

All computational cost analyses were performed on a system equipped with an Nvidia RTX Pro 6000 Blackwell Server Edition GPU[9] and a 12-core Intel(R) Xeon(R) Gold 5415+ CPU[10].

Across all sentiment analysis datasets, we observe that **MMD^2** offers the lowest end-to-end computational cost, closely followed by **MDM** and **PAD**, which also demonstrate strong efficiency. Notably, while Lens relies on LLM-based scoring—which substantially increases latency compared to embedding-based proxies—its runtime in the 7B setting (90.52 s/$\mathcal{D}$) is still considerably lower than that of Mauve (112.51 s/$\mathcal{D}$). This suggests that, although LLM scoring introduces additional overhead, Lens can remain competitive in throughput relative to existing generative evaluation methods when using smaller-scale LLMs.

C.7 Example Rubrics

Listing 3: Example characteristic descriptions $C_{\mathrm{s,r}}$

```
"Dataset B consistently specifies the analyst behind actions...",
"Dataset B maintains strict financial focus without political or
    entertainment tangents present in A...",
"Dataset B entries always directly connect stock movements to specific
    analyst actions...",
"Dataset B shows more frequent price target amount disclosures...",
"Dataset B uses standardized financial terminology consistently...",
"Dataset B maintains neutral tone in earnings reports...",
"Dataset B focuses exclusively on institutional analyst
    perspectives...",
"Dataset B headlines strictly follow '[Analyst] [Action] on [Ticker]
    [Rationale]' structure...",
"Dataset B contains no social media tags/hashtags...",
"Dataset B shows higher frequency of ETF coverage..."
```

Listing 4: Example characteristic descriptions $C_{\mathrm{r,s}}$

```
"Dataset B includes headlines without stock tickers ...",
"Dataset B contains non-financial news ...",
"Dataset B incorporates social media-style commentary...",
"Dataset B includes international/non-English company names...",
"Dataset B references non-institutional analysts/sources...",
"Dataset B features headlines about dividends...",
"Dataset B includes legal/regulatory actions unrelated to markets...",
"Dataset B uses technical trading jargon...",
"Dataset B contains macroeconomic commentary without stock links...",
```

[9] https://www.nvidia.com/en-us/data-center/rtx-pro-6000-blackwell-server-edition/

[10] https://www.intel.com/content/www/us/en/products/sku/232373/intel-xeon-gold-5415-processor-22-5m-cache-2-90-ghz/specifications.html

```
"Dataset B includes non-company-specific index/currency forecasts..."
```

C.8 LLM Usage

Model serving We use vLLM (Kwon et al., 2023) to serve open source models such as for data synthesis and dataset scoring. For `Llama3.3-70B-Instruct` model, we use Ollama [11] Q4_K quantized version to construct synthetic datasets for sentiment analysis. We use 2 * Nvidia A40 48GB GPUs for other models for synthesis and scoring. To improve LLMs' generation throughput, we use vLLM's batched inference feature and enable prefix-caching to further improve generation efficiency.

LLM hyperparameter For LENS rubric compilation and scoring, we set temperature to 0 and top_p to 0.95.

C.8.1 Data Synthesis Prompts

Listing 5: Zero-shot prompt used for sentiment analysis dataset generation.

```
Generate three realistic financial news headlines for sentiment analysis.

Guidelines for Generating Headlines:

Sentiment Labeling:
Each headline must be assigned a sentiment label based on its tone:
    -   Bearish (0): Indicates negative sentiment about a stock or market trend.
    -   Bullish (1): Indicates positive sentiment about a stock or market trend.
    -   Neutral (2): Indicates neutral or informational tone.

Now, generate three new financial news headlines following these guidelines.
    Please use JSON format and generate one type of each sentiment label (0, 1, 2)
    in your response.
```

Listing 6: Zero-shot with background knowledge prompt used for sentiment analysis dataset generation.

```
Generate three realistic financial news headlines about stock tickers following
    real-world financial reporting for sentiment analysis.

Guidelines for Generating Headlines:

1. Format & Style:
    -   Headlines must be concise and mimic real financial news.
    -   Use sentence case formatting (capitalize only the first word and proper
        nouns).
    -   Some headlines should start with a stock ticker (e.g., $AAPL -), while
        others should begin with the company name or a broader market trend.

2. Ticker Inclusion:
    -   At least one headline should include a stock ticker (e.g., $TSLA - or
        $NVDA -).
    -   Some headlines should refer to companies by name instead of tickers (e.g.,
         "Alphabet and Meta see price targets cut at Barclays").

3. Common Financial Themes:

Ensure headlines reflect realistic financial news topics,
including:
    -   Stock downgrades/upgrades
    -   Price target adjustments
    -   Market trends/economic outlook
    -   Company performance concerns
```

[11] https://ollama.com/

```
    -   Company news
    -   Company announcements
    -   Company events

4. Source Attribution:
    -   When relevant, mention an investment firm, analyst, or research group (e.g
        ., Morgan Stanley, Barclays, Oppenheimer).
    -   Do not fabricate research firms-use only well-known institutions.

5. Sentiment Labeling:

Each headline must be assigned a sentiment label based on its tone:
    -   Bearish (0): Indicates negative sentiment about a stock or market trend.
    -   Bullish (1): Indicates positive sentiment about a stock or market trend.
    -   Neutral (2): Indicates neutral or informational tone.

Sentiment Labeling:
Each headline must be assigned a sentiment label based on its tone:
    - Bearish (0): Indicates negative sentiment about a stock or market trend.
    - Bullish (1): Indicates positive sentiment about a stock or market trend.
    - Neutral (2): Indicates neutral or informational tone.

Now, generate three new financial news headlines following these guidelines.
    Please use JSON format and generate one type of each sentiment label (0, 1, 2)
    in your response.
```

Listing 7: Zero-shot with background and stock ticker information prompt used for sentiment analysis dataset generation.

```
Generate three realistic financial news headlines about stock tickers following
    real-world financial reporting for sentiment analysis.

Guidelines for Generating Headlines:

1. Format & Style:
    -   Headlines must be concise and mimic real financial news.
    -   Use sentence case formatting (capitalize only the first word and proper
        nouns).
    -   Some headlines should start with a stock ticker (e.g., $AAPL -), while
        others should begin with the company name or a broader market trend.

2. Ticker Inclusion:
    -   At least one headline should include a stock ticker (e.g., $TSLA - or
        $NVDA -).
    -   Some headlines should refer to companies by name instead of tickers (e.g.,
         "Alphabet and Meta see price targets cut at Barclays").

3. Common Financial Themes:

Ensure headlines reflect realistic financial news topics, including:
    -   Stock downgrades/upgrades
    -   Price target adjustments
    -   Market trends/economic outlook
    -   Company performance concerns
    -   Company news
    -   Company announcements
    -   Company events

4. Source Attribution:
    -   When relevant, mention an investment firm, analyst, or research group (e.g
        ., Morgan Stanley, Barclays, Oppenheimer).
    -   Do not fabricate research firms-use only well-known institutions.

5. Sentiment Labeling:
```

```
Each headline must be assigned a sentiment label based on its tone:
    -   Bearish (0): Indicates negative sentiment about a stock or market trend.
    -   Bullish (1): Indicates positive sentiment about a stock or market trend.
    -   Neutral (2): Indicates neutral or informational tone.

Now, generate three new financial news headlines about stock tickers: {
    stock_ticker} following these guidelines. Please use JSON format and generate
    one type of each sentiment label (0, 1, 2) for diversity.
```

C.8.2 Data Cleaning Prompts

Listing 8: Prompt used for filtering noisy synthetic images

```
You are a helpful assistant that filters out an image. You will be given an image
    and its corresponding text caption.

You should return true if the primary object in the image is not a ${label} in
    common sense. Return false otherwise.

Image:
{image}

Caption:
{caption}
```

C.8.3 Lens Rubric Compilation Prompts

Listing 9: Rubric compilation prompt used in sentiment analysis (commonalities)

```
You are a world class data analyst on financial news headline. You will be given
    some financial news headline samples from dataset A and dataset B. Based on the
     provided similar characteristics between them, list how B is similar to A.
    Return {num} points as a JSON list of strings. Please focus on specific and
    granular similarities between the two datasets, your generated characteristic
    points should apply to all the samples from the two datasets.

Samples from A:
{A}

Samples from B:
{B}
```

Listing 10: Rubric compilation prompt used in sentiment analysis (differences)

```
You are a world class data analyst on financial news headlines. You will be given
    some financial news headline samples from dataset A and dataset B. Based on the
     provided similar characteristics between them, list how B is {feedback} A.
    Please focus on granular differences between the two datasets, your generated
    characteristic points should apply to all the samples from the corresponding
    dataset (A or B). Return {num} points as a JSON list of strings.

Similar characteristics between A and B:
{similar_points}

Samples from A:
{A}

Samples from B:
{B}
```

Listing 11: Rubric compilation prompt used in Text2SQL (commonalities)

```
You are a world class data analyst on database queries in natural language. Given
    samples from dataset A and dataset B, list how B is {feedback} A. Return {num}
    points as a JSON list of strings. Please focus on specific and granular
    similarities between the two datasets, your generated characteristic points
    should apply to all the samples.

Question samples from A:
{A}

Question samples from B:
{B}
```

Listing 12: Rubric compilation prompt used in Text2SQL (differences)

```
You are a world class data analyst on database queries in natural language. Given
    query samples from dataset A and dataset B. Based on the provided similar
    characteristics between them, list how B is {feedback} A. Return {num} points
    as a JSON list of strings. Please focus on specific and granular differences
    between the two datasets, your generated characteristic points should apply to
    all the samples from the corresponding dataset (A or B).

Similar characteristics between A and B:
{similar_points}

Question samples from A:
{A}

Question samples from B:
{B}
```

Listing 13: Rubric compilation prompt used in image classification (commonalities)

```
Below are {num_image} images from dataset B:
{B}

Below are {num_images} images from dataset A:
{A}

Given some samples from image classification dataset A and dataset B, list how
    dataset B is similar to dataset A. Return ${num_points} points that summarize
    the similar characteristics of the two datasets. Focus on the characteristics
    of the image in terms of how they are structured, styled, or captured (e.g.,
    lighting, background, composition, etc.) rather than the image specifications
    such as resolution, size, etc. Your generated characteristic points should
    apply to all the samples from the corresponding dataset (A or B). Output should
     be a JSON list of strings.

Your listed points:
```

Listing 14: Rubric compilation prompt used in image classification (differences)

```
Below are {num_image} images from dataset B:
{B}

Below are {num_images} images from dataset A:
{A}

Given some samples from image classification dataset A and dataset B, list how
    dataset B is different from dataset A. Similar characteristics are provided
    below for reference. Return ${num_points} points that summarize the
    characteristics of the two datasets (e.g., dataset A is ... dataset B is ...).
    Focus on the characteristics of the images in terms of how they are structured,
     styled, or captured (e.g., lighting, background, composition, etc.) rather
    than the image specifications such as resolution, size, etc. Your generated
```

```
    characteristic points should apply to all the samples from the corresponding
    dataset (A or B). Output should be a JSON list of strings.

Similar characteristics:
${similar_characteristics}

Your listed points:
```

Listing 15: Rubric compilation prompt used in web navigation (commonalities)

```
Given some samples of web navigation tasks and the web accessibility tree of
    dataset A and dataset B, list how B is {feedback} A. Return {num} points that
    summarize the characteristics of the two datasets. The two accessibility trees
    generated from the same website are provided. Please only list out
    characteristics that are related to the proposed web navigation tasks, the
    accessibility trees are provided to only help you understand the context of the
     proposed tasks, therefore do not mention the accessibility tree in your
    response. Please focus on granular and specific characteristics, and your
    generated characteristic points should apply to all the samples. Output should
    be a JSON list of strings.

Accessibility Tree for dataset A:
{A_tree}

Sampled web navigation tasks from dataset A:
{A}

Accessibility Tree for dataset B:
{B_tree}

Sampled web navigation tasks from dataset B:
{B}
```

Listing 16: Rubric compilation prompt used in web navigation (differences)

```
Given some samples of proposed web navigation tasks and the web accessibility tree
     of dataset A and dataset B. Based on the similar characteristics between them.
     List how B is {feedback} A. Return {num} points that summarize the
    characteristics of the two datasets. The two accessibility trees of the same
    website are provided. Please only list out characteristics that are related to
    the proposed web navigation tasks, the accessibility trees are only provided to
     help you understand the context of the proposed tasks. Therefore do not list
    any characteristics that are related to the accessibility tree in your response
    . Please focus on granular and specific characteristics, and your generated
    characteristic points should apply to all samples in corresponding dataset (A
    or B). Output should be a JSON list of strings.

Similar characteristics between A and B:
{similar_points}

Accessibility Tree for dataset A:
{A_tree}

Sampled web navigation tasks from dataset A:
{A}

Accessibility Tree for dataset B:
{B_tree}

Sampled web navigation tasks from dataset B:
{B}
```

C.8.4 Lens Scoring Prompts

Listing 17: Scorer prompt used in sentiment analysis

```
You are given similarities and differences between two financial news headline
    datasets A and B.

Your task is to judge how likely is the given financial news headline comes from
    dataset {prediction}. Answer your judgement with one of the following strings:
    "very unlikely", "unlikely", "unsure", "likely", and "very likely".

Similar characteristics between dataset A and B:
{similar_characteristics}

Differences between dataset A and B:
{differences}

Financial news headline sample to be judged:
{example}

Your judgement in JSON format:
```

Listing 18: Scorer prompt used in Text2SQL

```
You are given similarities and differences between datasets A and B about database
     queries in natural language.

Your task is to judge how likely is the given database query in natural language
    comes from dataset {prediction}. Answer your judgement with one of the
    following strings: "very unlikely", "unlikely", "unsure", "likely", and "very
    likely".


Similar characteristics between dataset A and B:
{similar_characteristics}

Differences between dataset A and B:
{differences}

Natural language database query to be judged:
{example}

Your judgement in JSON format:
```

Listing 19: Scorer prompt used in image classification

```
You are given similarities and differences between datasets A and B.

Your task is to judge how likely is the given image comes from dataset {prediction
    }. Answer your judgement with one of the following strings: "very unlikely", "
    unlikely", "unsure", "likely", and "very likely".

Format:
{format_instructions}

Similar characteristics between dataset A and B:
{similar_characteristics}

Differences between dataset A and B:
{differences}

Image to be judged:
{image}

your judgement in JSON format:
```

Listing 20: Scorer prompt used in web navigation

```
You are given similar and different characteristics between two datasets A and B
    consisting of web navigation tasks.

Your objective is to judge how likely is the given web browsing task comes from
    dataset {prediction}. Answer your judgement with one of the following strings:
    "very unlikely", "unlikely", "unsure", "likely", and "very likely".

Format:
{format_instructions}

Similar characteristics between dataset A and B:
{similar_characteristics}

Different characteristics between dataset A and B:
{differences}

Web navigation task to be judged:
{example}

Your judgement in JSON format:
```