Let The Jury Decide: Fair Demonstration Selection for In-Context Learning through Incremental Greedy Evaluation

Anonymous ACL submission

Abstract

Large Language Models (LLMs) are powerful in-context learners, achieving strong performance with just a few high-quality demonstrations. However, fairness concerns arise in many in-context classification tasks, especially when predictions involve sensitive attributes. To address this, we propose **JUDGE**—a simple yet effective framework for selecting fair and representative demonstrations that improve group fairness in In-Context Learning. JUDGE constructs the demonstration set iteratively using a greedy approach, guided by a small, carefully selected *jury* set. Our method remains robust across varying LLM architectures and datasets, ensuring consistent fairness improvements. We evaluate JUDGE on *four* datasets using *four* LLMs, comparing it against *seven* baselines. Results show that JUDGE consistently improves fairness metrics without compromising accuracy.

1 Introduction

A key capability of Large Language Models (LLMs) is in-context learning (ICL) — the ability to learn from examples provided within a prompt, without requiring parameter updates (Brown et al., 2020; Dong et al., 2022). While research has advanced our understanding of ICL and techniques to enhance its effectiveness, a critical open question remains: how should we select fair and representative demonstration examples? This question becomes particularly critical in high-stakes domains where predictions directly impact human lives.

Consider a parole board using an LLM to assess recidivism risk. The model's predictions are shaped by the examples it is shown—if those examples reflect historical biases or overlook key rehabilitation factors, the system may produce plausible-looking predictions that perpetuate or amplify existing disparities. In sensitive domains like criminal justice, healthcare, and hiring, the selection of demonstrations directly influences both predictive reliability and equitable decision-making.

Existing demonstration selection strategies, with a few exceptions, largely focus on optimizing performance metrics such as accuracy (Peng et al., 2024; Wu et al., 2023). While these methods are effective for improving ICL performance, they often fail to account for fairness concerns. Parallel research has explored bias and fairness in LLM outputs (Gallegos et al., 2024) and their trustworthiness (Huang et al., 2024), but a key gap remains: how can we improve *group fairness* directly at the demonstration selection stage in in-context learning? Unlike prior work that dynamically selects demonstrations per test query (Wang et al., 2024), we explore an alternative: constructing a *single* demonstration set for an entire classification task.

In our work, we investigate several key questions. **Do different LLMs exhibit consistent fairness behavior across datasets?** We find significant variations in fairness outcomes across different LLMs, highlighting the need for adaptive approaches rather than one-size-fits-all solutions. **Do existing demonstration selection methods generalize across LLM architectures and datasets?** Our results reveal that most prior methods fail to maintain stable fairness improvements across different models due to inherent variability in LLM responses. **Can we design an effective, fairness-aware demonstration selection approach?** We propose a simple yet highly effective method, **JUDGE** (**JU**ry-based **D**emonstration Selection via **G**reedy **E**valuation)[1] that leverages each LLM's own predictions on a carefully curated set of *jury* examples to guide demonstration selection.

This paper makes several contributions. First, we provide a comprehensive analysis of existing approaches for fairness-aware demonstration selection across multiple datasets and architecures.

[1] Code: https://anonymous.4open.science/r/ACL25Code-6879

Second, we present **JUDGE**, a consistent and efficient framework for mining fair representative examples from large datasets for ICL. Third, we validate our approach through extensive empirical evaluation, showing significant fairness improvements without compromising accuracy across multiple fairness benchmarks. Finally, our systematic analysis demonstrates that the greedy construction approach is crucial for balancing fairness and accuracy, outperforming alternatives such as top-k selection and pooling-based methods. As LLMs continue to be deployed in increasingly sensitive domains, our work provides a practical framework for ensuring fairer outcomes while maintaining the efficiency that makes ICL attractive.

2 Preliminaries: Protected Groups, Attributes and Group Fairness

Protected groups are demographic subpopulations that should not face disparate treatment in model decisions. Let $\mathcal{G}$ denote the set of protected groups, each defined by a *protected attribute* such as race, gender, or age. For any instance x in dataset $\mathcal{D}$, its protected group membership is given by $g(x)$. The population is partitioned into distinct protected groups $\mathcal{G} = \{g_1, g_2, \ldots, g_l\}$, with each instance belonging to exactly one. For binary attributes (e.g., gender), this simplifies to $\mathcal{G} = \{g_1, g_2\}$.

Group fairness, or statistical fairness, aims to ensure that a model's behavior remains consistent across protected groups by enforcing that certain statistical measures are approximately equal across all protected groups, rather than focusing on individual-level fairness. We consider three established metrics: **Demographic Parity Difference** (ΔDP), which measures the absolute difference in positive rates between protected groups (Padh et al., 2021); **Equalized Odds Difference** (ΔEO), which measures the absolute difference in true positive and false positive rates between groups (Hardt et al., 2016); and **Mutual Information** (MI) (Kamishima et al., 2012; Anahideh et al., 2022), which quantifies the mutual information between protected attributes and selection decisions (Details in Appendix A).

3 Proposed Approach

3.1 Problem Formulation: Fairest Prompt Search for In-Context Learning

Given a large language model M and an input x, ICL makes predictions by conditioning on a demonstration set $\mathcal{S} = \{(x_1, y_1), ..., (x_k, y_k)\}$. The model processes these demonstrations along with the query input as:

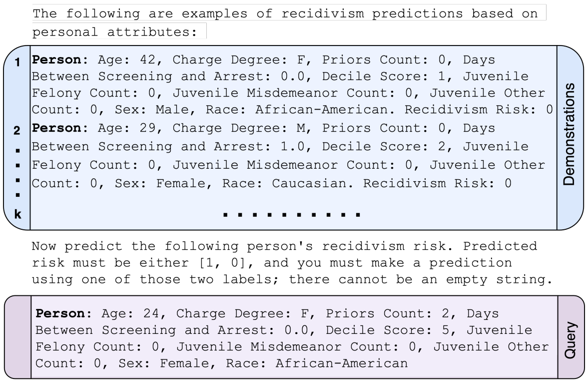


Figure 1: Example ICL Prompt on the COMPAS dataset

$$\text{prompt}(\mathcal{S}, x) = [(x_1, y_1), ..., (x_k, y_k), x] \quad (1)$$

Let $\mathcal{X}$ denote the input space, $\mathcal{Y}$ the label space, and $\mathcal{L}$ the natural language space. The formatting function ϕ maps k input-label pairs and the query to a natural language prompt as seen in Figure 1:

$$\phi : \underbrace{(\mathcal{X} \times \mathcal{Y})^k}_{\text{k demonstration pairs}} \times \underbrace{\mathcal{X}}_{\text{query input}} \rightarrow \mathcal{L} \quad (2)$$

$$\phi(\text{prompt}(\mathcal{S}, x)) \in \mathcal{L} \quad (3)$$

The model then predicts:

$$\hat{y} = \text{argmax}_{y \in \mathcal{Y}} M(y|\phi(\text{prompt}(\mathcal{S}, x))) \quad (4)$$

where $M(y|\phi(\text{prompt}(\mathcal{S}, x)))$ represents the model's predicted probability distribution over the label space $\mathcal{Y}$, which we denote for brevity as:

$$\hat{y} = M(\mathcal{S}, x) \quad (5)$$

The fundamental challenge is selecting an effective and fair demonstration set $\mathcal{S}$ from a large candidate pool. For a pool of size $|\mathcal{D}|$ and desired demonstration set size k, there are $\binom{|\mathcal{D}|}{k}$ possible combinations. For even modest values like $|\mathcal{D}| = 1000$ and $k = 5$, this yields over 8 trillion possible demonstration sets, making exhaustive search intractable. Fairness constraints further complicate this selection.

Our approach, JUDGE addresses fair demonstration selection through a multi-step process as shown in Figure 2. Let:

Train set $\mathcal{D}_{train}$: The pool of available examples, where each example $x \in \mathcal{D}_{train}$ has associated features, a label $y \in [0, 1]$, and protected group membership g(x).

Jury set $\mathcal{J}$: A small curated subset of examples extracted from $\mathcal{D}_{train}$ that serves to evaluate the fairness and effectiveness of candidate demonstration sets.

Candidate set $\mathcal{D}_{candidate}$: The complement of the jury set with respect to the train set, defined as $\mathcal{D}_{candidate} = \mathcal{D}_{train} \setminus \mathcal{J}$, from which potential demonstrations can be selected.

Reduced candidate set $\mathcal{D}_{reduced} \subseteq \mathcal{D}_{candidate}$: A pruned subset of the candidate set, selected to maintain semantic diversity while reducing computational complexity. Demonstrations are selected from this subset.

Protected groups $\mathcal{G} = \{g_1, g_2, \ldots, g_l\}$: The set of groups defined by protected attributes, where each example belongs to exactly one group. We consider a binary setting where $\mathcal{G} = \{g_1, g_2\}$.

Selected set $\mathcal{S} \subseteq \mathcal{D}_{reduced}$: The chosen subset of k examples that will serve as demonstrations, where k is typically small (e.g., 5-10) due to context length constraints.

Our *objective* is to find a demonstration set $\mathcal{S}^*$ that optimizes both predictive accuracy (a) and fairness (f):

$$\mathcal{S}^* = \text{argmax}_{\mathcal{S} \subseteq \mathcal{D}_{reduced}, |\mathcal{S}|=k} \, \text{score}(\mathcal{S}, \mathcal{J}) \qquad (6)$$

$$\text{score}(\mathcal{S}, \mathcal{J}) = \omega \cdot f(\mathcal{S}, \mathcal{J}) + (1 - \omega) \cdot a(\mathcal{S}, \mathcal{J}) \qquad (7)$$

The accuracy, a term measures the model's predictive performance on the jury set:

$$a(\mathcal{S}, \mathcal{J}) = \frac{1}{|\mathcal{J}|} \sum_{(x,y) \in \mathcal{J}} \mathbb{I}[M(\mathcal{S}, x) = y] \qquad (8)$$

For the fairness, $f(\mathcal{S}, \mathcal{J})$ term, we use the widely used demographic parity difference (detailed in Appendix A) to assess the demonstration set's fairness using the jury set:

$$\begin{aligned} f(\mathcal{S}, \mathcal{J}) = -|P(M(\mathcal{S}, x) = 1 \mid g(x) = g_1) - \\ P(M(\mathcal{S}, x) = 1 \mid g(x) = g_2)| \end{aligned} \qquad (9)$$

Note that we negate the demographic parity difference since lower differences indicate better fairness, allowing both accuracy and fairness terms to be maximized in the same direction in Equation 6.

To summarise, JUDGE consists of three main steps. First, we construct a balanced and diverse jury set $\mathcal{J}$ which evaluates candidate examples based on both fairness metrics and predictive performance. This jury set is drawn from the training set and subsequently removed to form the candidate pool. Next, we prune the candidate pool to maximize semantic diversity and limit computational overhead. Finally, we employ a greedy selection algorithm that iteratively builds the demonstration set $\mathcal{S}$ by adding, at each step, a demonstration from $\mathcal{D}_{reduced}$ that maximizes the fairness-accuracy objective (Equation 7) over the jury set $\mathcal{J}$.

3.2 Jury Set Composition

The jury set $\mathcal{J}$ is carefully constructed to ensure balanced representation across all protected groups and labels. We define all possible group-label pairs as $\mathcal{C} = \{(g, y) : g \in \mathcal{G}, y \in \mathcal{Y}\}$. For example, in a binary setting where g represents gender (Male, Female) and y represents income level (>50k as 1, $\leq$ 50k as 0), we have $\mathcal{C} = \{(\text{Female}, 0), (\text{Female}, 1), (\text{Male}, 0), (\text{Male}, 1$

Each subset $\mathcal{J}_{g,y}$ consists of $m = |\mathcal{J}|/|\mathcal{C}|$ examples, selected to maximize semantic diversity.

For each example x, we compute an embedding $e(x)$ using SentenceBERT (Reimers, 2019). We measure the semantic similarity between examples using cosine similarity:

$$\text{sim}(x_i, x_j) = \frac{e(x_i) \cdot e(x_j)}{\|e(x_i)\| \|e(x_j)\|} \qquad (10)$$

To construct a diverse subset $\mathcal{J}_{g,y}$, we iteratively select the next example x_{next} that minimizes its maximum similarity to the previously selected examples.

$$x_{\text{next}} = \arg \min_{x_i \notin \mathcal{J}_{g,y}} \max_{x_j \in \mathcal{J}_{g,y}} \text{sim}(x_i, x_j) \qquad (11)$$

Therefore, the subset $\mathcal{J}_{g,y}$ is calculated as:

$$\begin{aligned} &\mathcal{J}_{g,y} = \{x_1, ..., x_m\} \text{ where} \\ &\quad x_i = \arg \min_{x \in \mathcal{D}_{g,y} \setminus \{x_1, \ldots, x_{i-1}\}} \max_{j<i} \text{sim}(x, x_j) \end{aligned} \qquad (12)$$

where $\mathcal{D}_{g,y}$ represents the subset of examples in $\mathcal{D}_{train}$ with protected group g and label y. The final jury set is the union of these diverse subsets:

$$\mathcal{J} = \bigcup_{(g,y) \in \mathcal{C}} \mathcal{J}_{g,y} \qquad (13)$$

3.3 Diversity-Based Candidate Pruning

To efficiently reduce the size of the candidate pool while preserving coverage across the semantic space, we employ a selection strategy based on semantic similarity.

We construct the reduced set $\mathcal{D}_{reduced}$ iteratively by selecting examples that are maximally distinct from those already chosen, following Sec 3.2. Given a target size n , the selection is defined as:

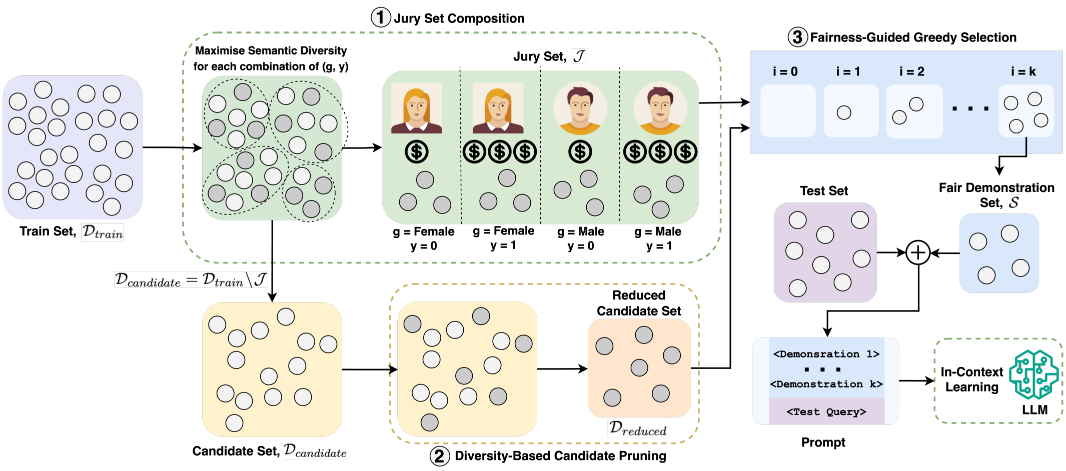


Figure 2: JUDGE consists of three main steps: (1) constructing a balanced and diverse jury set $\mathcal{J}$ (2) pruning the candidate pool to reduce computational overhead, and (3) iteratively selecting demonstrations using a greedy algorithm that optimizes a weighted combination of fairness and accuracy scores over the jury set.

$$\mathcal{D}_{reduced} = \{x_1, ..., x_n\} \text{ where}$$
$$x_i = \arg \min_{x \in \mathcal{D}_{candidate} \setminus \{x_1,...,x_{i-1}\}} \max_{j<i} \text{sim}(x, x_j) \tag{14}$$

This selection process ensures that the final subset $\mathcal{D}_{reduced}$ preserves the semantic diversity of the original pool while being computationally tractable for subsequent operations.

3.4 Fairness-Guided Greedy Selection

The algorithm constructs the demonstration set $\mathcal{S}$ iteratively using a greedy selection process, optimizing both fairness and accuracy over the jury set $\mathcal{J}$. At each iteration t , the example that maximizes the marginal improvement in the overall score is added to $\mathcal{S}$.

The process starts with an empty set, $\mathcal{S}_0 = \emptyset$. At $t = 1$, each example in $\mathcal{D}_{reduced}$ is evaluated independently as the first demonstration, and the one yielding the highest fairness-accuracy score on the jury set is selected as x_1, and $\mathcal{S}_1 = \mathcal{S}_0 \cup \{x_1\}$. At $t = 2$, we evaluate each of the remaining candidates in $\mathcal{D}_{\text{reduced}} \setminus \mathcal{S}_1$ in combination with x_1, forming two-example demonstration sets, selecting x_2 that maximizes the score and $\mathcal{S}_2 = \mathcal{S}_1 \cup \{x_2\}$. This process continues until $t = k$.

Formally, starting with an empty set $\mathcal{S}_0 = \emptyset$, at each iteration t until $|\mathcal{S}_t| = k$, we select:

$$x_t = \text{argmax}_{x \in \mathcal{D}_{reduced} \setminus \mathcal{S}_{t-1}} \text{score}(\mathcal{S}_{t-1} \cup \{x\}, \mathcal{J}) \tag{15}$$

where $\mathcal{S}_t = \mathcal{S}_{t-1} \cup \{x_t\}$ and score is computed as defined in Section 3.1.

While this greedy approach does not guarantee finding the globally optimal demonstration set, it offers several advantages. First, it reduces search complexity from $\binom{|\mathcal{D}_{reduced}|}{k}$ to $O(k|\mathcal{D}_{reduced}|)$, drastically reducing the search space. Second, it ensures interpretability, as each demonstration is chosen based on a clear improvement metric. Finally, by evaluating candidates based on their marginal contribution, it captures interaction effects, leading to a more effective and fair selection.

Our approach is detailed in Algorithm 1. The pseudocode for the helper function DiverseSelect, which is based on the description from Section 3.2, can be found in Algorithm 2 in the Appendix.

4 Complexity Analysis

The complexity is dominated by **LLM inference**. In JUDGE, each demonstration in $\mathcal{D}_{reduced}$ is evaluated with every jury member to compute demographic parity and accuracy. Since this is repeated k times to build a k-sized set, the overall complexity is $O(k \cdot |\mathcal{D}_{\text{reduced}}| \cdot |\mathcal{J}|)$. Unlike our method, **exhaustive search** evaluates all possible subsets of size k from N demonstrations, i.e., a complexity of $O(N^K)$ which is infeasible for large N and k. A detailed complexity comparison with other baselines can be found in Appendix C.

5 Results

5.1 Datasets

We use *four* widely studied fairness datasets across different domains and protected attributes (details in Appendix B.2). **Adult Income** (Dua and Graff,

Algorithm 1 JUDGE

Require: Training set $\mathcal{D}_{train}$, protected groups $\mathcal{G}$, labels $\mathcal{Y}$, desired size k, jury size per group m, candidate pool size n, trade-off ω
Ensure: Fair demonstration set $\mathcal{S}$

```
// Step 1: Construct balanced jury set
C ← {(g, y) : g ∈ G, y ∈ Y}
J ← ∅
for (g, y) ∈ C do
    D_{g,y} ← {x ∈ D_train : g(x) = g ∧ label(x) = y}
    J_{g,y} ← DiverseSelect(D_{g,y}, m)
    J ← J ∪ J_{g,y}
end for
// Step 2: Prune candidate pool
D_reduced ← DiverseSelect(D_train \ J, n)
// Step 3: Greedy selection
S_0 ← ∅
for t ← 1 to k do
    x_t ← None, s_max ← −∞
    for x ∈ D_reduced \ S_{t−1} do
        S_temp ← S_{t−1} ∪ {x}
        f ← f(S_temp, J), a ← a(S_temp, J)
        s ← ω · f + (1 − ω) · a
        if s > s_max then
            x_t ← x, s_max ← s
        end if
    end for
    S_t ← S_{t−1} ∪ {x_t}
end for
return S_k
```

2019) to predict whether income exceeds \$50K (protected attribute: *gender*). **COMPAS** (Angwin et al., 2016) to predict recidivism risk (protected attribute: *race*). **Law School (LSAC)** (Wightman, 1998) to predict whether a student passes the bar (protected attribute: *race*). **ACS Income** (Ding et al., 2021) to predict whether income exceeds \$50K (protected attribute: *gender*).

5.2 Language Models

To assess the generalizability of JUDGE, we evaluate our approach using *four* open-source language models of varying parameters from different sources: Meta's **LLaMA-3** 8B (Dubey et al., 2024), Mistral AI's **Mistral** 7B (Jiang et al., 2023), Google's **Gemma-2** 9B (Riviere et al., 2024), and Alibaba's **Qwen-2.5** 32B (Hui et al., 2024).

5.3 Baselines

We compare our approach against *seven* baseline methods for demonstration selection. **Random** selects k demonstrations randomly from the training set. **Balanced** employs stratified random sampling to maintain equal representation across protected groups and label. **Counterfactual** (Li et al., 2023) selects from privileged groups and generates counterfactual examples by flipping sensitive attributes while preserving other features. **Instruct** (Atwood et al., 2024) guides the model toward fairness via explicit prompt instructions. **FCG** (Hu et al., 2024) uses clustering and evolutionary strategies to curate diverse, representative demonstrations while considering fairness metrics. **FairICL** (Bhaila et al., 2024) leverages latent concept variables to evaluate demonstration fairness and guide selection, learning fair concepts from training data to promote fairness while maintaining utility. **FADS** (Wang et al., 2024) implements a two-stage filtering approach (data and model bias mitigation) followed by similarity-based selection with balanced representation across groups and labels. Unlike adaptive methods like **FADS**, which select demonstrations per test instance, **JUDGE** selects a *single fixed set* for all test examples. We evaluate our method against both *fixed* and *adaptive* approaches.

5.4 Experimental Setup

For each dataset-model combination, we conduct experiments with two demonstration set sizes: $k = 5$ and $k = 10$, using 20% of the data for testing where standard splits are not provided. We evaluate performance using four metrics: accuracy (Acc.), Demographic Parity Difference (ΔDP), Equalized Odds Difference (ΔEO), and mutual information (MI), as defined in Section 2. All results reported in Tables 1-8 show the mean of 3 reproduction runs. For space constraints, results for $k = 10$ (Tables 5-8) are provided in Appendix B.3. The reduced candidate set $\mathcal{D}_{reduced}$ is pruned to *3% of* $\mathcal{D}_{candidate}$ via semantic diversity maximization. Jury set sizes are $m = 25$ for Adult and COMPAS, and $m = 50$ for Law School and ACS, with selection details in Appendix B.11. A sensitivity analysis on ω is provided in Appendix B.6.

5.4.1 Intrinsic Fairness Differences Among LLMs and Datasets

We note an interesting pattern across our results: *different LLMs report significantly different fairness metrics*. This is evident when examining the *Random* baseline. For instance, with $k = 5$ on Adult (Table 1), Gemma-2 produces a ΔDP score of 0.394, compared to LLaMA-3's 0.185, more than twice the disparity in demographic parity. These variations persist across datasets, with Gemma-2 often exhibiting greater unfairness, e.g., ΔDP = 0.310 on COMPAS, compared to Mistral's 0.097 (Table 2), over three times the value.

Perhaps less surprisingly, datasets themselves

Table 1: Results for Adult with 5 demonstrations, across 4 LLMs. Each cell shows $Mean_{S.D.}$

Model	Method	Acc. ↑	ΔDP ↓	ΔEO ↓	MI ↓
LLAMA-3-8B	Random	$0.772_{0.008}$	$0.185_{0.004}$	$0.191_{0.006}$	$0.023_{0.002}$
	Balanced	$0.706_{0.015}$	$0.216_{0.011}$	$0.146_{0.014}$	$0.022_{0.001}$
	Cfact.	$0.731_{0.017}$	$0.185_{0.019}$	$0.158_{0.023}$	$0.018_{0.003}$
	Instruct	$0.753_{0.013}$	$0.299_{0.011}$	$0.308_{0.012}$	$0.052_{0.006}$
	FairICL	$0.764_{0.009}$	$0.170_{0.004}$	$0.097_{0.008}$	$0.016_{0.002}$
	FCG	$0.795_{0.011}$	$0.097_{0.009}$	$0.157_{0.006}$	$0.011_{0.001}$
	FADS	$0.743_{0.015}$	$0.157_{0.012}$	$0.114_{0.014}$	$0.019_{0.003}$
	JUDGE	$\mathbf{0.798}_{0.012}$	$\mathbf{0.078}_{0.011}$	$\mathbf{0.049}_{0.012}$	$\mathbf{0.004}_{0.001}$
MISTRAL-7B	Random	$0.709_{0.013}$	$0.201_{0.010}$	$0.124_{0.009}$	$0.019_{0.003}$
	Balanced	$0.594_{0.014}$	$0.230_{0.011}$	$0.185_{0.012}$	$0.025_{0.004}$
	Cfact.	$0.722_{0.011}$	$0.143_{0.008}$	$0.193_{0.013}$	$0.011_{0.003}$
	Instruct	$0.729_{0.021}$	$0.162_{0.019}$	$0.171_{0.023}$	$0.015_{0.004}$
	FairICL	$0.761_{0.006}$	$0.151_{0.011}$	$0.159_{0.007}$	$0.012_{0.002}$
	FCG	$0.752_{0.015}$	$0.132_{0.014}$	$0.093_{0.019}$	$\mathbf{0.006}_{0.001}$
	FADS	$\mathbf{0.769}_{0.009}$	$0.180_{0.008}$	$0.129_{0.005}$	$0.021_{0.002}$
	JUDGE	$0.767_{0.012}$	$\mathbf{0.101}_{0.009}$	$\mathbf{0.024}_{0.005}$	$\mathbf{0.006}_{0.001}$
GEMMA-2-9B	Random	$0.754_{0.006}$	$0.394_{0.008}$	$0.423_{0.013}$	$0.091_{0.005}$
	Balanced	$0.701_{0.014}$	$0.482_{0.023}$	$0.413_{0.026}$	$0.113_{0.021}$
	Cfact.	$0.752_{0.015}$	$0.311_{0.015}$	$0.372_{0.011}$	$0.087_{0.016}$
	Instruct	$0.742_{0.011}$	$0.428_{0.009}$	$0.479_{0.013}$	$0.108_{0.008}$
	FairICL	$0.753_{0.014}$	$0.318_{0.019}$	$0.392_{0.026}$	$0.089_{0.013}$
	FCG	$0.755_{0.017}$	$0.233_{0.025}$	$0.192_{0.018}$	$\mathbf{0.013}_{0.003}$
	FADS	$0.759_{0.013}$	$0.353_{0.011}$	$0.387_{0.016}$	$0.072_{0.006}$
	JUDGE	$\mathbf{0.769}_{0.012}$	$\mathbf{0.177}_{0.018}$	$\mathbf{0.101}_{0.009}$	$0.018_{0.003}$
QWEN-2.5-32B	Random	$0.745_{0.012}$	$0.215_{0.010}$	$0.132_{0.010}$	$0.023_{0.004}$
	Balanced	$0.708_{0.014}$	$0.245_{0.013}$	$0.165_{0.012}$	$0.027_{0.003}$
	Cfact.	$0.748_{0.014}$	$0.225_{0.014}$	$0.143_{0.011}$	$0.025_{0.003}$
	Instruct	$0.733_{0.007}$	$0.239_{0.013}$	$0.161_{0.009}$	$0.026_{0.005}$
	FairICL	$0.743_{0.009}$	$0.192_{0.012}$	$0.147_{0.015}$	$0.027_{0.009}$
	FCG	$0.762_{0.013}$	$0.111_{0.014}$	$0.098_{0.013}$	$0.007_{0.002}$
	FADS	$0.712_{0.009}$	$0.220_{0.007}$	$0.141_{0.006}$	$0.023_{.003}$
	JUDGE	$\mathbf{0.771}_{0.008}$	$\mathbf{0.096}_{0.005}$	$\mathbf{0.062}_{0.004}$	$\mathbf{0.005}_{0.001}$

Table 2: Results for COMPAS with 5 demonstrations, across 4 LLMs. Each cell shows $Mean_{S.D.}$

Model	Method	Acc. ↑	ΔDP ↓	ΔEO ↓	MI ↓
LLAMA-3-8B	Random	$0.617_{0.011}$	$0.209_{0.009}$	$0.199_{0.008}$	$0.021_{0.003}$
	Balanced	$0.620_{0.012}$	$0.235_{0.011}$	$0.218_{0.013}$	$0.027_{0.002}$
	Cfact.	$0.582_{0.009}$	$0.187_{0.006}$	$0.193_{0.007}$	$0.017_{0.001}$
	Instruct	$0.566_{0.010}$	$0.135_{0.009}$	$0.164_{0.010}$	$0.015_{0.001}$
	FairICL	$0.621_{0.009}$	$0.192_{0.007}$	$0.188_{0.006}$	$0.020_{0.002}$
	FCG	$0.614_{0.007}$	$0.182_{0.005}$	$0.197_{0.005}$	$0.019_{0.001}$
	FADS	$0.575_{0.008}$	$0.167_{0.006}$	$0.160_{0.005}$	$0.014_{0.002}$
	JUDGE	$\mathbf{0.656}_{0.010}$	$\mathbf{0.105}_{0.008}$	$\mathbf{0.082}_{0.007}$	$\mathbf{0.006}_{0.001}$
MISTRAL-7B	Random	$0.513_{0.012}$	$0.097_{0.008}$	$0.120_{0.009}$	$0.016_{0.002}$
	Balanced	$0.512_{0.007}$	$0.079_{0.005}$	$0.083_{0.004}$	$0.013_{0.003}$
	Cfact.	$0.487_{0.010}$	$0.059_{0.009}$	$\mathbf{0.062}_{0.009}$	$0.015_{0.004}$
	Instruct	$0.497_{0.012}$	$0.082_{0.010}$	$0.105_{0.008}$	$0.014_{0.002}$
	FairICL	$0.515_{0.006}$	$0.082_{0.005}$	$0.098_{0.005}$	$0.017_{0.004}$
	FCG	$0.489_{0.009}$	$0.074_{0.004}$	$0.108_{0.006}$	$0.013_{0.003}$
	FADS	$0.531_{0.010}$	$0.091_{0.005}$	$0.117_{0.007}$	$0.015_{0.009}$
	JUDGE	$\mathbf{0.541}_{0.007}$	$\mathbf{0.055}_{0.004}$	$0.075_{0.004}$	$\mathbf{0.002}_{0.000}$
GEMMA-2-9B	Random	$0.615_{0.008}$	$0.310_{0.005}$	$0.314_{0.006}$	$0.049_{0.003}$
	Balanced	$0.601_{0.009}$	$0.359_{0.006}$	$0.348_{0.005}$	$0.067_{0.004}$
	Cfact.	$0.604_{0.007}$	$0.261_{0.004}$	$0.272_{0.005}$	$0.044_{0.005}$
	Instruct	$0.609_{0.011}$	$0.291_{0.009}$	$0.309_{0.012}$	$0.047_{0.006}$
	FairICL	$0.622_{0.010}$	$0.265_{0.011}$	$0.282_{0.012}$	$0.040_{0.005}$
	FCG	$0.648_{0.007}$	$0.099_{0.003}$	$0.091_{0.005}$	$0.008_{0.003}$
	FADS	$0.621_{0.014}$	$0.307_{0.011}$	$0.303_{0.09}$	$0.053_{0.009}$
	JUDGE	$\mathbf{0.665}_{0.006}$	$\mathbf{0.062}_{0.002}$	$\mathbf{0.039}_{0.003}$	$\mathbf{0.002}_{0.000}$
QWEN-2.5-32B	Random	$0.637_{0.007}$	$0.242_{0.005}$	$0.221_{0.006}$	$0.029_{0.003}$
	Balanced	$\mathbf{0.652}_{0.008}$	$0.248_{0.007}$	$0.240_{0.011}$	$0.031_{0.005}$
	Cfact.	$0.611_{0.008}$	$0.244_{0.006}$	$0.228_{0.006}$	$0.031_{0.004}$
	Instruct	$0.633_{0.006}$	$0.234_{0.003}$	$0.214_{0.004}$	$0.026_{0.002}$
	FairICL	$0.639_{0.008}$	$0.211_{0.005}$	$0.218_{0.005}$	$0.025_{0.003}$
	FCG	$0.623_{0.006}$	$0.149_{0.004}$	$0.144_{0.003}$	$0.018_{0.003}$
	FADS	$0.645_{0.008}$	$0.224_{0.006}$	$0.207_{0.004}$	$0.025_{0.003}$
	JUDGE	$0.649_{0.004}$	$\mathbf{0.138}_{0.005}$	$\mathbf{0.134}_{0.003}$	$\mathbf{0.010}_{0.001}$

vary in fairness, with the same model reporting very different fairness metrics across different datasets. More interestingly, *certain baselines behave dramatically differently across models*. For example, $Instruct$ achieves strong fairness on Law School with Mistral for both $k = 5$ and 10 , yet completely sacrifices fairness on Qwen-2.5B and Gemma-2, despite maintaining high accuracy (Table 4, 8). One trend remains consistent: *methods behave similarly across demonstration sizes*, with performance staying stable across $k = 5$ and $k = 10$ for a given model, dataset, and method.

5.5 Performance Comparison

JUDGE consistently outperforms baselines across 32 settings (4 LLMs * 4 Datasets * 2 Demonstration set sizes), achieving the best performance in most cases and near-best results in the instances where it is not the top performer. We attribute this to its greedy approach, which iteratively selects demonstrations by maximizing their marginal contribution based on LLM feedback using a semantically diverse jury set. Given the high variability in data types and LLM architectures, we believe this step-by-step feedback is key to generalizability.

Notably, baselines that incorporate LLM feedback, like FCG, tend to perform better than those relying solely on heuristics, which often lack consistency—excelling in some cases but failing in others. For instance, Counterfactual selection significantly improves fairness over Random on Gemma-2 for Adult, but worsens on Qwen-2.5 for the same dataset (Table 1). Similarly, Instruct improves fairness over Random on LLaMA-3 for COMPAS (Table 2) but significantly harms it on Adult using the same LLM. FADS, designed to mitigate both model and data bias, performs well in many cases but struggles on certain datasets. FairICL, which trains a local LLaMA model to rank demonstrations, suffers from limited generalizability due to architectural differences between models. Overall, JUDGE remains the most consistent across all settings, improving fairness across metrics while maintaining accuracy. Its LLM-driven, stepwise construction ensures robust, data- and model-agnostic performance, making it a stronger, more reliable approach than existing baselines.

5.6 Ablation: Greedy vs. Top-k Selection

To validate our greedy selection approach, we compare it against two alternatives: (1) Top-k, the top k candidates that individually perform the best on

Table 3: Results for ACS with 5 demonstrations, across 4 LLMs. Each cell shows $Mean_{S.D.}$

Model	Method	Acc.↑	ΔDP↓	ΔEO↓	MI↓
LLAMA-3-8B	Random	$0.693_{0.009}$	$0.122_{0.008}$	$0.106_{0.009}$	$0.008_{0.001}$
	Balanced	$0.689_{0.016}$	$0.089_{0.009}$	$0.070_{0.008}$	$0.004_{0.000}$
	Cfact.	$0.653_{0.005}$	$0.092_{0.004}$	$0.092_{0.004}$	$0.004_{0.000}$
	Instruct	$0.684_{0.010}$	$0.115_{0.006}$	$0.101_{0.006}$	$0.008_{0.001}$
	FairICL	$0.688_{0.011}$	$0.098_{0.008}$	$0.010_{0.004}$	$0.008_{0.002}$
	FCG	$0.759_{0.010}$	$0.066_{0.005}$	$0.071_{0.004}$	$0.002_{0.006}$
	FADS	$0.697_{0.008}$	$0.116_{0.006}$	$0.101_{0.004}$	$0.008_{0.001}$
	JUDGE	$\mathbf{0.764}_{0.007}$	$\mathbf{0.045}_{0.002}$	$\mathbf{0.049}_{0.003}$	$\mathbf{0.001}_{0.000}$
MISTRAL-7B	Random	$0.603_{0.007}$	$0.091_{0.005}$	$0.052_{0.006}$	$0.005_{0.001}$
	Balanced	$0.558_{0.013}$	$0.070_{0.009}$	$\mathbf{0.032}_{0.009}$	$0.003_{0.000}$
	Cfact.	$0.607_{0.009}$	$0.085_{0.005}$	$0.063_{0.004}$	$0.006_{0.001}$
	Instruct	$0.592_{0.017}$	$0.094_{0.12}$	$0.108_{0.011}$	$0.007_{0.001}$
	FairICL	$0.610_{0.005}$	$0.089_{0.004}$	$0.051_{0.003}$	$0.005_{0.001}$
	FCG	$0.648_{0.008}$	$0.051_{0.007}$	$0.069_{0.004}$	$0.007_{0.001}$
	FADS	$0.599_{0.012}$	$0.088_{0.006}$	$0.051_{0.004}$	$0.005_{0.000}$
	JUDGE	$\mathbf{0.651}_{0.011}$	$\mathbf{0.031}_{0.005}$	$0.036_{0.006}$	$\mathbf{0.001}_{0.000}$
GEMMA-2-9B	Random	$0.696_{0.009}$	$0.225_{0.007}$	$0.223_{0.010}$	$0.028_{0.004}$
	Balanced	$0.707_{0.012}$	$0.233_{0.009}$	$0.179_{0.009}$	$0.028_{0.002}$
	Cfact.	$0.690_{0.010}$	$0.227_{0.008}$	$0.227_{0.009}$	$0.027_{0.003}$
	Instruct	$0.696_{0.016}$	$0.263_{0.010}$	$0.278_{0.010}$	$0.039_{0.005}$
	FairICL	$0.691_{0.014}$	$0.211_{0.007}$	$0.218_{0.012}$	$0.027_{0.004}$
	FCG	$0.705_{0.012}$	$0.141_{0.007}$	$0.136_{0.009}$	$0.018_{0.002}$
	FADS	$\mathbf{0.709}_{0.016}$	$0.205_{0.006}$	$0.274_{0.010}$	$0.031_{0.003}$
	JUDGE	$0.704_{0.010}$	$\mathbf{0.131}_{0.006}$	$\mathbf{0.124}_{0.006}$	$\mathbf{0.013}_{0.001}$
QWEN-2.5-32B	Random	$0.727_{0.014}$	$0.101_{0.008}$	$0.059_{0.010}$	$0.005_{0.001}$
	Balanced	$0.728_{0.012}$	$0.076_{0.008}$	$\mathbf{0.017}_{0.008}$	$0.003_{0.000}$
	Cfact.	$0.731_{0.005}$	$0.087_{0.003}$	$0.032_{0.003}$	$0.004_{0.001}$
	Instruct	$0.735_{0.014}$	$0.191_{0.009}$	$0.125_{0.010}$	$0.018_{0.002}$
	FairICL	$0.724_{0.011}$	$0.091_{0.006}$	$0.076_{0.007}$	$0.005_{0.001}$
	FCG	$0.727_{0.006}$	$0.059_{0.003}$	$0.051_{0.003}$	$0.002_{0.000}$
	FADS	$0.729_{0.003}$	$0.097_{0.002}$	$0.046_{0.004}$	$0.005_{0.001}$
	JUDGE	$\mathbf{0.739}_{0.010}$	$\mathbf{0.025}_{0.005}$	$0.036_{0.005}$	$\mathbf{0.001}_{0.000}$

Table 4: Results for Law School with 5 demonstrations, across 4 LLMs. Each cell shows $Mean_{S.D.}$

Model	Method	Acc.↑	ΔDP↓	ΔEO↓	MI↓
LLAMA-3-8B	Random	$0.895_{0.012}$	$0.299_{0.009}$	$0.493_{0.015}$	$0.054_{0.004}$
	Balanced	$0.663_{0.016}$	$0.406_{0.008}$	$0.377_{0.006}$	$0.047_{0.005}$
	Cfact.	$0.871_{0.015}$	$0.272_{0.010}$	$0.435_{0.018}$	$0.044_{0.003}$
	Instruct	$0.862_{0.019}$	$0.197_{0.011}$	$0.307_{0.019}$	$0.032_{0.003}$
	FairICL	$0.764_{0.015}$	$0.312_{0.008}$	$0.346_{0.006}$	$0.045_{0.002}$
	FCG	$0.909_{0.016}$	$0.082_{0.016}$	$0.178_{0.020}$	$0.019_{0.003}$
	FADS	$0.898_{0.004}$	$0.242_{0.003}$	$0.353_{0.006}$	$0.039_{0.001}$
	JUDGE	$\mathbf{0.911}_{0.026}$	$\mathbf{0.057}_{0.027}$	$\mathbf{0.104}_{0.035}$	$\mathbf{0.005}_{0.001}$
MISTRAL-7B	Random	$0.905_{0.009}$	$0.187_{0.011}$	$0.338_{0.007}$	$0.029_{0.003}$
	Balanced	$0.871_{0.012}$	$0.219_{0.004}$	$0.362_{0.007}$	$0.031_{0.001}$
	Cfact.	$0.904_{0.012}$	$0.200_{0.011}$	$0.418_{0.008}$	$0.029_{0.003}$
	Instruct	$0.913_{0.010}$	$\mathbf{0.023}_{0.004}$	$0.077_{0.005}$	$\mathbf{0.004}_{0.000}$
	FairICL	$0.902_{0.010}$	$0.173_{0.006}$	$0.311_{0.009}$	$0.026_{0.003}$
	FCG	$0.943_{0.014}$	$0.038_{0.006}$	$0.091_{0.04}$	$0.018_{0.002}$
	FADS	$0.934_{0.006}$	$0.103_{0.004}$	$0.227_{0.003}$	$0.019_{0.002}$
	JUDGE	$\mathbf{0.946}_{0.013}$	$0.027_{0.003}$	$\mathbf{0.059}_{0.004}$	$0.008_{0.001}$
GEMMA-2-9B	Random	$0.853_{0.011}$	$0.372_{0.007}$	$0.569_{0.010}$	$0.058_{0.004}$
	Balanced	$0.756_{0.007}$	$0.419_{0.011}$	$0.436_{0.008}$	$0.056_{0.003}$
	Cfact.	$0.747_{0.007}$	$0.366_{0.004}$	$0.358_{0.006}$	$0.042_{0.003}$
	Instruct	$\mathbf{0.878}_{0.004}$	$0.344_{0.005}$	$0.553_{0.003}$	$0.056_{0.001}$
	FairICL	$0.844_{0.010}$	$0.341_{0.009}$	$0.357_{0.012}$	$0.041_{0.002}$
	FCG	$0.845_{0.013}$	$0.258_{0.009}$	$0.267_{0.011}$	$0.029_{0.003}$
	FADS	$0.877_{0.009}$	$0.287_{0.006}$	$0.502_{0.007}$	$0.047_{0.003}$
	JUDGE	$0.855_{0.013}$	$\mathbf{0.227}_{0.009}$	$\mathbf{0.214}_{0.008}$	$\mathbf{0.025}_{0.003}$
QWEN-2.5-32B	Random	$0.865_{0.007}$	$0.327_{0.005}$	$0.414_{0.004}$	$0.052_{0.002}$
	Balanced	$0.831_{0.011}$	$0.392_{0.005}$	$0.493_{0.006}$	$0.061_{0.002}$
	Cfact.	$0.840_{0.009}$	$0.366_{0.006}$	$0.418_{0.008}$	$0.055_{0.005}$
	Instruct	$0.883_{0.008}$	$0.370_{0.005}$	$0.534_{0.006}$	$0.072_{0.004}$
	FairICL	$0.860_{0.018}$	$0.316_{0.010}$	$0.449_{0.014}$	$0.057_{0.002}$
	FCG	$0.862_{0.016}$	$0.248_{0.013}$	$0.293_{0.012}$	$0.035_{0.003}$
	FADS	$\mathbf{0.889}_{0.021}$	$0.238_{0.012}$	$0.419_{0.016}$	$0.044_{0.009}$
	JUDGE	$0.882_{0.016}$	$\mathbf{0.214}_{0.013}$	$\mathbf{0.273}_{0.015}$	$\mathbf{0.027}_{0.001}$

the jury set, (2) Top-k-Balanced, which is a stratified selection that picks the top samples from each combination of protected group and label, (g, y). Results show that greedy selection consistently outperforms both methods, highlighting the importance of marginal contribution of each example in building a fair and effective demonstration set. The effect of Top-k and Top-k-Balanced varies by dataset. As shown in Figure 3, on Adult (LLaMA-3-8B), Top-k exhibits a dramatic drop in fairness performance, while Top-k-Balanced fares better. On COMPAS, we see competitive fairness performance across variants, but upon closer inspection we observe that Top-k and Top-k-Balanced selection suffers large drops in accuracy. These findings underscore the inherent variability in ICL and reinforce the strength of the greedy approach, which incrementally selects candidates while considering their interactions with the existing set.

5.7 Impact of Jury Set Size

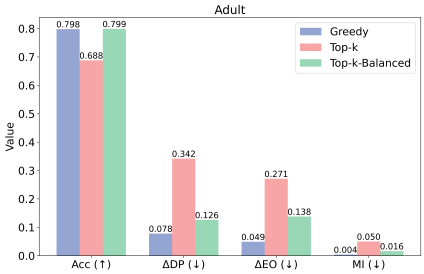
To assess the impact of jury size, we vary the number of examples per group-label combination m from 1 to 100, keeping all other parameters constant with $k = 5$ demonstrations. Figure 4 (LLaMA-3 on Adult) shows results for accuracy and ΔDP (full results in Appendix B.4, Figure 10). Results on *Adult* indicate that performance stabilizes as m increases, with diminishing returns beyond $m > 25$ despite higher computational costs. Accuracy plateaus quickly, with $m = 5$ or 10 being sufficient, while fairness improves up to $m = 50$.

5.8 Jury Set Diversity

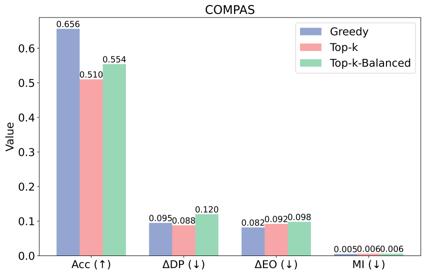
To examine the impact of semantic diversity in jury set construction, we compare three methods: (1) Random Sampling, (2) Random-Balanced (random sampling after enforcing equal representation across protected group-label combinations), and (3) Semantic Diversity-based selection. We fix $m = 25$ for this comparison. With jury sets constrained to be small for computational efficiency, Figure 5 shows that diversity-based selection outperforms both alternatives on the ACS dataset. A similar experiment on Adult is provided in Appendix B.5.

6 Related Work

Demonstration Selection in ICL The problem of selecting demonstrations for ICL has received significant attention. (Liu et al., 2022) showed



(a) Adult dataset



(b) COMPAS dataset

Figure 3: Comparison of Greedy vs. Top-k alternatives

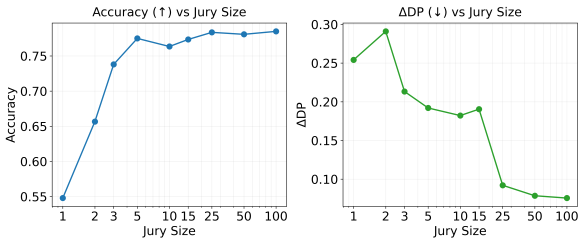


Figure 4: Accuracy and ΔDP against the size of the jury set for Adult. Higher sizes show diminishing returns.

that finding demonstrations which are semantically similar to the test data often shows promising results. Wu et al. (2023) addressed this challenge by establishing a select-then-rank framework where they first limit the search space of demonstrations and rank the remaining examples through heuristics. Peng et al. (2024) highlighted that both data and model factors contribute to variability in performance. Meanwhile, Ma et al. (2023) showed that predictive performance can be improved by selecting examples that minimize predictive bias. To address efficiency concerns, Yang et al. (2023) proposed a two-stage Determinantal Point Process (DPP) method to select a fixed, representative subset of demonstrations, improving efficiency while maintaining performance.

Fair Demonstration Selection in ICL The fairness of language models has received significant attention (Doan et al., 2024; Chu et al., 2024). Liu et al. (2024) showed that LLMs exhibit significant bias in tabular classification. In ICL, fair demonstration selection is crucial. Hu et al. (2024) investigated how varying the composition of demonstrations affects fairness outcomes in ICL. The authors proposed a fairness-aware selection method that employs clustering and evolutionary strategies to curate a diverse and representative sample set from the training data. Meanwhile, Wang et al. (2024) introduced FADS, which addresses the challenge of fair demonstration selection by mitigating both model bias and bias in the data. Other approaches have explored leveraging counterfactual analysis. Bhaila et al. (2024) introduced a method that uses latent concept variables learned through counterfactual examples to evaluate the fairness of demonstrations. The idea of utilizing counterfactual examples is also presented by Li et al. (2023), which picks examples from the privileged group and flips the sensitive attribute to create new examples. Finally, Atwood et al. (2024) showed how prompting the model by explicitly asking it to be fair can also be effective. However, existing approaches have limitations, as shown in our experiments, while our proposed method, JUDGE, ensures more consistent and reliable fairness improvements.

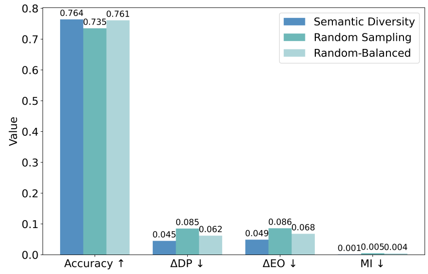


Figure 5: Comparison of diversity vs. other sampling techniques for the jury set on the ACS dataset.

7 Conclusion

We propose JUDGE, a greedy framework for fair demonstration selection in ICL, guided by a jury set. Across four datasets and four LLM architectures, our method consistently improves fairness while maintaining accuracy, outperforming existing approaches. We further highlight the high variability of different methods across different datasets and language models, and establish the importance of considering demonstrations as a cohesive set rather than as individual examples to ensure fairness. As LLMs expand into critical applications, JUDGE offers a practical and robust solution for ensuring fairness in ICL.

Limitations

This work investigates the problem of fairness aware demonstration selection for in-context learning. In order to do so, this work explores various open-source LLM architectures from Google, Meta, Mistral, and Alibaba. While these architectures have varied sizes ranging from 7B to 32B parameters, a key limitation in our work is that, due to hardware limitations we do not investigate the effect on truly massive models like LLAMA-3-405B. Furthermore, financial constraints prevent us from using closed-source paid platforms like GPT-4o, given the large number of LLM queries required across our datasets, baselines, LLMs and demonstration sizes. Nonetheless, we believe we chose a diverse and representative set of highly performant open-source LLMs to make our study comprehensive. Furthermore, our study limits itself to exploring binary in-context classification as well as binary sensitive group settings. In the future, we plan to consider broader classification settings. Finally, in line with prior work, we aimed to conduct a comprehensive study across widely popular fairness datasets, which are typically tabular in nature and are thus serialized into a natural language prompt for the LLM. In the future, we hope to study other types of data in the context of fairness in large language models.

References

Hadis Anahideh, Abolfazl Asudeh, and Saravanan Thirumuruganathan. 2022. Fair active learning. *Expert Systems with Applications*, 199:116981.

Julia Angwin, Jeff Larson, Surya Mattu, and Lauren Kirchner. 2016. Machine bias: There's software used across the country to predict future criminals. and it's biased against blacks. *ProPublica*, 23(2016):77–91.

James Atwood, Preethi Lahoti, Ananth Balashankar, Flavien Prost, and Ahmad Beirami. 2024. Inducing group fairness in llm-based decisions. *arXiv preprint arXiv:2406.16738*.

Karuna Bhaila, Minh-Hao Van, Kennedy Edemacu, Chen Zhao, Feng Chen, and Xintao Wu. 2024. Fair in-context learning via latent concept variables. *arXiv preprint arXiv:2411.02671*.

Tom B Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, et al. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901.

Zhibo Chu, Zichong Wang, and Wenbin Zhang. 2024. Fairness in large language models: A taxonomic survey. *ACM SIGKDD explorations newsletter*, 26(1):34–48.

Frances Ding, Moritz Hardt, John Miller, and Ludwig Schmidt. 2021. Retiring adult: New datasets for fair machine learning. In *Advances in Neural Information Processing Systems*, volume 34, pages 6478–6490.

Thang Viet Doan, Zichong Wang, Nhat Nguyen Minh Hoang, and Wenbin Zhang. 2024. Fairness in large language models in three hours. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management*, CIKM '24, page 5514–5517, New York, NY, USA. Association for Computing Machinery.

Qingxiu Dong, Lei Li, Damai Dai, Ce Zheng, Jingyuan Ma, Rui Li, Heming Xia, Jingjing Xu, Zhiyong Wu, Tianyu Liu, et al. 2022. A survey on in-context learning. *arXiv preprint arXiv:2301.00234*.

Dheeru Dua and Casey Graff. 2019. Uci machine learning repository.

Anirudh Dubey et al. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Isabel O Gallegos, Ryan A Rossi, Joe Barrow, Md Mehrab Tanjim, Sungchul Kim, Franck Dernoncourt, Tong Yu, Ruiyi Zhang, and Nesreen K Ahmed. 2024. Bias and fairness in large language models: A survey. *Computational Linguistics*, pages 1–79.

Moritz Hardt, Eric Price, and Nati Srebro. 2016. Equality of opportunity in supervised learning. *Advances in neural information processing systems*, 29.

Jingyu Hu, Weiru Liu, and Mengnan Du. 2024. Strategic demonstration selection for improved fairness in llm in-context learning. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 7460–7475.

Yue Huang, Lichao Sun, Haoran Wang, Siyuan Wu, Qihui Zhang, Yuan Li, Chujie Gao, et al. 2024. Position: TrustLLM: Trustworthiness in large language models. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pages 20166–20270. PMLR.

B. Hui et al. 2024. Qwen2.5-coder technical report. *arXiv preprint arXiv:2409.12186*.

A. Q. Jiang et al. 2023. Mistral 7b. *arXiv preprint arXiv:2310.06825*.

Toshihiro Kamishima, Shotaro Akaho, Hideki Asoh, and Jun Sakuma. 2012. Fairness-aware classifier with prejudice remover regularizer. In *Machine Learning and Knowledge Discovery in Databases: European Conference, ECML PKDD 2012, Bristol, UK, September 24-28, 2012. Proceedings, Part II 23*, pages 35–50. Springer.

Yunqi Li, Lanjing Zhang, and Yongfeng Zhang. 2023. Fairness of chatgpt. *arXiv preprint arXiv:2305.18569*.

Jiachang Liu, Dinghan Shen, Yizhe Zhang, Bill Dolan, Lawrence Carin, and Weizhu Chen. 2022. What makes good in-context examples for GPT-3? In *Proceedings of Deep Learning Inside Out (DeeLIO 2022): The 3rd Workshop on Knowledge Extraction and Integration for Deep Learning Architectures*, pages 100–114, Dublin, Ireland and Online. Association for Computational Linguistics.

Yanchen Liu, Srishti Gautam, Jiaqi Ma, and Himabindu Lakkaraju. 2024. Confronting LLMs with traditional ML: Rethinking the fairness of large language models in tabular classifications. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 3603–3620, Mexico City, Mexico. Association for Computational Linguistics.

Huan Ma, Changqing Zhang, Yatao Bian, Lemao Liu, Zhirui Zhang, Peilin Zhao, Shu Zhang, Huazhu Fu, Qinghua Hu, and Bingzhe Wu. 2023. Fairness-guided few-shot prompting for large language models. *Advances in Neural Information Processing Systems*, 36:43136–43155.

Kirtan Padh, Diego Antognini, Emma Lejal-Glaude, Boi Faltings, and Claudiu Musat. 2021. Addressing fairness in classification with a model-agnostic multi-objective algorithm. In *Uncertainty in artificial intelligence*, pages 600–609. PMLR.

Keqin Peng, Liang Ding, Yancheng Yuan, Xuebo Liu, Min Zhang, Yuanxin Ouyang, and Dacheng Tao. 2024. Revisiting demonstration selection strategies in in-context learning. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 9090–9101, Bangkok, Thailand. Association for Computational Linguistics.

N Reimers. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. *arXiv preprint arXiv:1908.10084*.

Morgane Riviere et al. 2024. Gemma 2: Improving open language models at a practical size. *arXiv preprint arXiv:2408.00118*.

Song Wang, Peng Wang, Yushun Dong, Tong Zhou, Lu Cheng, Yangfeng Ji, and Jundong Li. 2024. On demonstration selection for improving fairness in language models. In *Workshop on Socially Responsible Language Modelling Research*.

Linda F Wightman. 1998. Lsac national longitudinal bar passage study. Technical report, Law School Admission Council.

Zhiyong Wu, Yaoxiang Wang, Jiacheng Ye, and Lingpeng Kong. 2023. Self-adaptive in-context learning: An information compression perspective for in-context example selection and ordering. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1423–1436, Toronto, Canada. Association for Computational Linguistics.

Zhao Yang, Yuanzhe Zhang, Dianbo Sui, Cao Liu, Jun Zhao, and Kang Liu. 2023. Representative demonstration selection for in-context learning with two-stage determinantal point process. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 5443–5456, Singapore. Association for Computational Linguistics.

A Fairness Metrics Formulation

Here we provide detailed mathematical formulations of the fairness metrics used in our analysis. For all metrics, we take absolute values to ensure positive measures of disparity, where zero indicates perfect fairness and larger values indicate greater disparity.

A.1 Demographic Parity Difference (ΔDP)

The Demographic Parity Difference (ΔDP) measures the absolute difference in positive label rates between groups:

$$\Delta DP = |P(y=1 \mid g(x)=g_1) - P(y=1 \mid g(x)=g_2)| \tag{16}$$

where $y = 1$ denotes a positive label. A ΔDP of 0 indicates perfect demographic parity.

A.2 Equalized Odds Difference (ΔEO)

The Equalized Odds Difference (ΔEO) measures disparities in both true positive rates (TPR) and false positive rates (FPR) between groups:

$$\Delta EO = \max\left(|\text{TPR}_{g_1} - \text{TPR}_{g_2}|, |\text{FPR}_{g_1} - \text{FPR}_{g_2}|\right) \tag{17}$$

where

$$\text{TPR}_g = P(y=1 \mid g(x)=g, y^*=1) \tag{18}$$

$$\text{FPR}_g = P(y=1 \mid g(x)=g, y^*=0) \tag{19}$$

Here, y^* represents the true label, and $y = 1$ represents the predicted positive label. A ΔEO of 0 indicates perfect equalized odds.

A.3 Mutual Information Fairness

The mutual information between protected group membership G and the positive label assignment is:

$$I(G;Y) = \sum_{g,y} P(g,y) \log \frac{P(g,y)}{P(g)P(y)} \tag{20}$$

where y denotes whether an instance receives a positive label. Lower mutual information indicates greater independence between the positive label assignment and protected group membership. This metric is naturally non-negative, with 0 indicating perfect independence.

B Additional Experiment Details

B.1 DiverseSelect

The pseudocode for maximizing semantic diversity in selection is shown in Algorithm 2.

Algorithm 2 DiverseSelect: Diversity-Based Example Selection

Require: Initial pool $\mathcal{D}$, target size k
Ensure: Diverse subset $\mathcal{D}_{diverse}$

Compute $Sij = \text{sim}(x_i, x_j)$ for all $x_i, x_j \in \mathcal{D}$
$\mathcal{D}_{diverse} \leftarrow x_r$ where x_r is randomly sampled from $\mathcal{D}$
for $t \leftarrow 1$ to $k - 1$ **do**
  for $x_i \in \mathcal{D} \setminus \mathcal{D}_{diverse}$ **do**
    $s_i \leftarrow \max_{x_j \in \mathcal{D}_{diverse}} Sij$
  end for
  $x_t \leftarrow \arg\min_{x_i \in \mathcal{D} \setminus \mathcal{D}_{diverse}} s_i$
  $\mathcal{D}_{diverse} \leftarrow \mathcal{D}_{diverse} \cup x_t$
end for
return $\mathcal{D}_{diverse}$

The following are examples of recidivism predictions based on personal attributes:

Demonstrations

1 **Person**: Age: 42, Charge Degree: F, Priors Count: 0, Days Between Screening and Arrest: 0.0, Decile Score: 1, Juvenile Felony Count: 0, Juvenile Misdemeanor Count: 0, Juvenile Other Count: 0, Sex: Male, Race: African-American. Recidivism Risk: 0
2 **Person**: Age: 29, Charge Degree: M, Priors Count: 0, Days Between Screening and Arrest: 1.0, Decile Score: 2, Juvenile Felony Count: 0, Juvenile Misdemeanor Count: 0, Juvenile Other Count: 0, Sex: Female, Race: Caucasian. Recidivism Risk: 0
k ....

Now predict the following person's recidivism risk. Predicted risk must be either [1, 0], and you must make a prediction using one of those two labels; there cannot be an empty string.

Query

Person: Age: 24, Charge Degree: F, Priors Count: 2, Days Between Screening and Arrest: 0.0, Decile Score: 5, Juvenile Felony Count: 0, Juvenile Misdemeanor Count: 0, Juvenile Other Count: 0, Sex: Female, Race: African-American

Figure 6: COMPAS Prompt Template

You are a classifier, predicting whether or not a student will pass (label 1) or fail (label 0) the bar based on student attributes. Now, the following are examples of bar exam outcomes based on student attributes:

Demonstrations

1 **Student**: LSAT Score: 39.0, Undergraduate GPA: 3.6, Law School GPA: 0.82, Decile Rank: 9.0, Family Income: 3.0, Full-time Status: 1.0, Part-time Status: 0.0, School Tier: 4.0, Cluster: 1.0, Index 60/40: 817.894717, Year of Birth: 68.0, Gender: female, Race: Black.
Bar Exam Outcome: 1
2 **Student**: LSAT Score: 39.0, Undergraduate GPA: 3.5, Law School GPA: -1.57, Decile Rank: 1.0, Family Income: 4.0, Full-time Status: 1.0, Part-time Status: 0.0, School Tier: 4.0, Cluster: 1.0, Index 60/40: 807.894717, Year of Birth: 68.0, Gender: male, Race: white.
Bar Exam Outcome: 1
k ....

Now predict the following student's bar exam outcome. Predicted bar exam outcome must be either [1, 0], and you must make a prediction using one of those two labels; there cannot be an empty string.

Query

Student: LSAT Score: 42.0, Undergraduate GPA: 3.3, Law School GPA: 0.49, Decile Rank: 9.0, Family Income: 5.0, Full-time Status: 1.0, Part-time Status: 0.0, School Tier: 3.0, Cluster: 3.0, Index 60/40: 835.263136, Year of Birth: 68.0, Gender: male, Race: white.
Bar Exam Outcome Prediction:

Figure 7: Law School Prompt Template

B.2 Dataset Details

Adult Income The UCI Adult dataset (Dua and Graff, 2019) contains demographic and employment information for 48,842 individuals. The task is to predict whether annual income exceeds $50,000, with gender as the protected attribute. The prompt template for this dataset is shown in Figure 8.

COMPAS This dataset (Angwin et al., 2016) includes criminal history and demographic data for 7,214 defendants. The classification task is predicting recidivism risk. We use a binarized race (Caucasian vs African-American) as the protected attribute. The prompt template for this dataset is shown in Figure 6.

Law School The LSAC dataset (Wightman, 1998) contains admissions data and academic performance for over 22,000 law school students. The model predicts whether a student passes the bar, with a binarized race (Caucasian vs Not-Caucasian) as the protected attribute. The prompt template for this dataset is shown in Figure 7.

ACS Income The ACS PUMS dataset (Ding et al., 2021) contains demographic and employment information from the American Community Survey. The task predicts if income exceeds $50,000, using gender as the protected attribute. The ACS Income dataset in its original form contains over 1.66 million datapoints, which is far larger than all other datasets that we consider, combined. For LLM in-context classification, this becomes prohibitively expensive from a computation perspective. As a result, we randomly downsample ACS Income down to 48,842 samples, which is the same size as the closely related Adult Dataset. Both datasets track American income data, but ACS provides much newer information from 2018 instead of 1994 for Adult. The prompt template for this dataset is shown in Figure 9.

```
The following are examples of income classification based on
personal attributes:

1  Person: Age: 65, Workclass:  Local-gov, Education:  12th,
   Number of Years of Education: 8, Marital Status:  Widowed,
   Occupation:  Exec-managerial, Relationship:  Not-in-family,
   Race:  White, Gender:  Male, Capital Gain: 2009, Capital
   Loss: 0, Hours Per Week: 44, Native Country:  United-States.
   Income: 0
2  Person: Age: 31, Workclass:  Private, Education:  Some-
   college, Number of Years of Education: 10, Marital Status:
   Married-civ-spouse, Occupation:  Farming-fishing,
   Relationship:  Husband, Race:  White, Gender:  Male, Capital
   Gain: 7298, Capital Loss: 0, Hours Per Week: 50, Native
   Country:  United-States.
k  Income: 1
   ..........
                                                  [Demonstrations]

Now predict the following person's income. Predicted Income
must be either [1, 0], and you must make a prediction using one
of those two labels, there cannot be an empty string.

   Person: Age: 35, Workclass:  Self-emp-inc, Education:
   Bachelors, Number of Years of Education: 13, Marital Status:
   Married-civ-spouse, Occupation:  Exec-managerial,
   Relationship:  Husband, Race:  White, Gender:  Male, Capital
   Gain: 0, Capital Loss: 0, Hours Per Week: 60, Native Country:
   United-States.
   Income Prediction:
                                                  [Query]
```

Figure 8: Adult Prompt Template

```
The following are examples of income classification based on
personal attributes:

1  Person: Age: 60.0, Workclass: employee of a private for-
   profit company, Education: regular high school diploma,
   Marital Status: never married or under 15 years old,
   Occupation: Hairdressers, Hairstylists, And Cosmetologists,
   Place of Birth: Pennsylvania, Relationship: reference person,
   Hours Worked Per Week: 9.0, Gender: Male, Race: White alone.
   Income: 0
2  Person: Age: 64.0, Workclass: employee of a private for-
   profit company, Education: master's degree, Marital Status:
   married, Occupation: Other Managers, Place of Birth: Indiana,
   Relationship: reference person, Hours Worked Per Week: 45.0,
   Gender: Female, Race: White alone.
k  Income: 1
   ..........
                                                  [Demonstrations]

Now predict the following person's income. Predicted Income
must be either [1, 0], and you must make a prediction using one
of those two labels, there cannot be an empty string.

   Person: Age: 49.0, Workclass: employee of a private for-profit
   company, Education: bachelor's degree, Marital Status:
   married, Occupation: Chief Executives And Legislators, Place
   of Birth: Texas, Relationship: reference person, Hours Worked
   Per Week: 40.0, Gender: Male, Race: White alone.
   Income Prediction:
                                                  [Query]
```

Figure 9: ACS Prompt Template

Table 5: Results for Adult with 10 demonstrations. Each cell shows $Mean_{S.D.}$

	Method	Acc. ↑	ΔDP ↓	ΔEO ↓	MI ↓
LLAMA-3-8B	Random	$0.779_{0.004}$	$0.133_{0.003}$	$0.118_{0.004}$	$0.017_{0.001}$
	Balanced	$0.751_{0.013}$	$0.221_{0.019}$	$0.137_{0.022}$	$0.025_{0.002}$
	Cfact.	$0.776_{0.018}$	$0.144_{0.014}$	$0.142_{0.015}$	$0.015_{0.003}$
	Instruct	$0.781_{0.022}$	$0.252_{0.017}$	$0.289_{0.019}$	$0.046_{0.004}$
	FairICL	$0.777_{0.015}$	$0.146_{0.012}$	$0.164_{0.135}$	$0.014_{0.003}$
	FCG	$0.788_{0.017}$	$0.189_{0.014}$	$0.163_{0.017}$	$0.023_{0.003}$
	FADS	$0.772_{0.013}$	$0.161_{0.011}$	$0.098_{0.005}$	$0.020_{0.003}$
	JUDGE	$\mathbf{0.794}_{0.011}$	$\mathbf{0.082}_{0.009}$	$\mathbf{0.092}_{0.008}$	$\mathbf{0.008}_{0.001}$
MISTRAL-7B	Random	$0.755_{0.014}$	$0.209_{0.012}$	$0.262_{0.009}$	$0.023_{0.004}$
	Balanced	$0.585_{0.009}$	$0.220_{0.011}$	$0.170_{0.008}$	$0.023_{0.003}$
	Cfact.	$0.731_{0.014}$	$0.141_{0.016}$	$0.090_{0.016}$	$0.010_{0.003}$
	Instruct	$0.742_{0.014}$	$0.182_{0.013}$	$0.212_{0.018}$	$0.012_{0.005}$
	FairICL	$0.763_{0.011}$	$0.143_{0.008}$	$0.155_{0.006}$	$0.013_{0.002}$
	FCG	$0.758_{0.022}$	$0.122_{0.013}$	$0.083_{0.016}$	$0.012_{0.003}$
	FADS	$\mathbf{0.775}_{0.012}$	$0.192_{0.009}$	$0.244_{0.013}$	$0.022_{0.003}$
	JUDGE	$0.771_{0.010}$	$\mathbf{0.010}_{0.012}$	$\mathbf{0.058}_{0.010}$	$\mathbf{0.010}_{0.001}$
GEMMA-2-9B	Random	$0.774_{0.009}$	$0.365_{0.005}$	$0.484_{0.012}$	$0.090_{0.006}$
	Balanced	$0.721_{0.015}$	$0.389_{0.022}$	$0.455_{0.029}$	$0.116_{0.015}$
	Cfact.	$0.762_{0.020}$	$0.276_{0.017}$	$0.383_{0.019}$	$0.072_{0.013}$
	Instruct	$0.764_{0.013}$	$0.408_{0.013}$	$0.523_{0.011}$	$0.107_{0.009}$
	FairICL	$0.763_{0.013}$	$0.301_{0.021}$	$0.323_{0.024}$	$0.072_{0.011}$
	FCG	$0.778_{0.011}$	$0.176_{0.013}$	$\mathbf{0.179}_{0.014}$	$0.053_{0.003}$
	FADS	$0.766_{0.010}$	$0.378_{0.009}$	$0.414_{0.016}$	$0.087_{0.007}$
	JUDGE	$\mathbf{0.792}_{0.010}$	$\mathbf{0.173}_{0.015}$	$0.197_{0.019}$	$\mathbf{0.047}_{0.005}$
QWEN-2.5-32B	Random	$0.741_{0.016}$	$0.210_{0.015}$	$0.129_{0.004}$	$0.021_{0.003}$
	Balanced	$0.728_{0.017}$	$0.223_{0.019}$	$0.152_{0.013}$	$0.026_{0.004}$
	Cfact.	$0.743_{0.011}$	$0.219_{0.010}$	$0.135_{0.009}$	$0.025_{0.002}$
	Instruct	$0.715_{0.009}$	$0.236_{0.010}$	$0.157_{0.011}$	$0.025_{0.002}$
	FairICL	$0.756_{0.012}$	$0.204_{0.010}$	$0.151_{0.010}$	$0.025_{0.003}$
	FCG	$\mathbf{0.778}_{0.011}$	$0.128_{0.010}$	$0.099_{0.007}$	$0.011_{0.003}$
	FADS	$0.706_{0.010}$	$0.206_{0.007}$	$0.132_{0.005}$	$0.022_{0.004}$
	JUDGE	$0.775_{0.009}$	$\mathbf{0.101}_{.008}$	$\mathbf{0.078}_{.004}$	$\mathbf{0.007}_{.001}$

B.3 Additional Results

This section presents results for all LLMs and all datasets with 10 demonstrations provided for In-Context Learning. These can be seen in Tables 5-8 for each of the four datasets.

B.4 Full Results for the Effect of Jury Set Size

Here we provide the results for all metrics for our experiment in Section 5.7, which tests the effect of different jury set sizes on the Adult dataset using LLAMA-3-8B. This is shown in Figure 10. This figure additionally shows the ΔEO and MI metrics which show the same pattern as the ΔDP metric.

B.5 Additional Jury Set Diversity Experiment

We conduct the same jury set diversity experiment from Section 5.8 on the Adult dataset, comparing random selection, balanced-random selection, and our diversity-based approach. As with ACS, the jury set size is fixed at $m = 25$ across all methods.

Figure 11 illustrates that diversity-based selection also outperforms other sampling strategies on the Adult dataset, reinforcing the importance of semantic diversity in jury construction.

B.6 Sensitivity Analysis

To understand the impact of trade-off parameters in different methods, we conduct a sensitivity analysis by varying the fairness-accuracy balancing coefficients in JUDGE, FCG, and FairICL on the Adult dataset over LLAMA-3-8B. We select FCG and FairICL as baselines because their respective authors explicitly identify α and $\tilde{D}$ as key parameters that influence fairness, making them well-suited for comparison with JUDGE.

B.7 JUDGE: Sensitivity to ω

JUDGE introduces ω as a parameter that controls the trade-off between accuracy and fairness. The selection of demonstrations is influenced by ω, where lower values prioritize fairness while higher val-

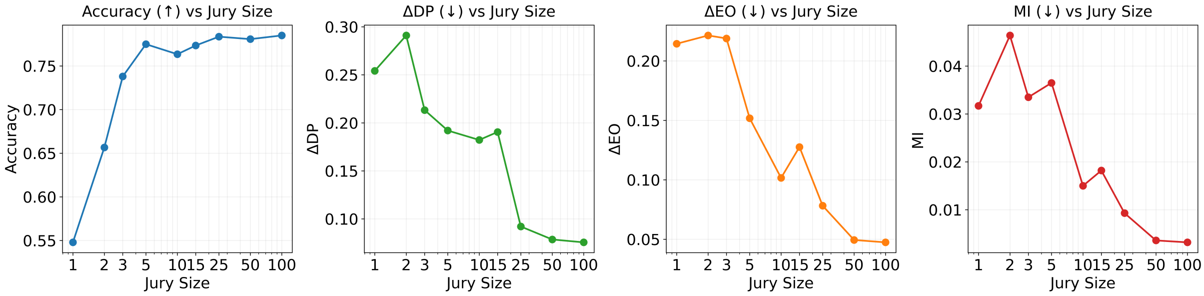


Figure 10: Comparing metrics against the size of the jury set for Adult. Higher sizes show diminishing returns.

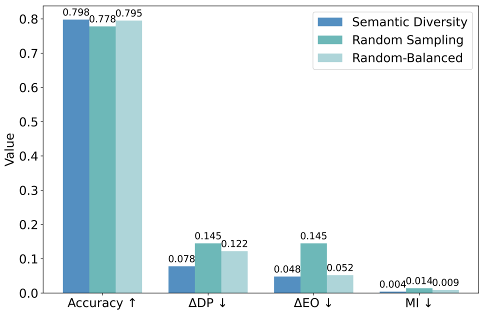


Figure 11: Comparison of diversity vs. other sampling techniques for the jury set on the Adult dataset.

ues emphasize accuracy. We evaluate JUDGE at $\omega \in \{0.4, 0.5, 0.6, 0.7, 0.8\}$. We choose this set because we find in our experiments that ω values that prioritize fairness slightly more than accuracy work well, improving fairness while also retaining predictive performance.

B.8 FCG: Sensitivity to α

FCG uses α in the EvolScore function to balance accuracy and fairness. The original paper sets $\alpha = 0.5$, and we analyze values in $\{0.3, 0.4, 0.5, 0.6, 0.7\}$ to assess how the fairness-accuracy trade-off shifts.

B.9 FairICL: Sensitivity to $\tilde{D}$

FairICL introduces $\tilde{D}$, which represents the fraction of augmented data used for fairness-aware training. The original study evaluates FairICL at $\tilde{D} \in \{0\%, 25\%, 50\%, 100\%\}$, highlighting its influence on fairness. To provide a more fine-grained analysis, we add an additional evaluation at $\tilde{D} = 75\%$, resulting in the set $\{0\%, 25\%, 50\%, 75\%, 100\%\}$.

B.10 Results: Accuracy vs. Fairness Trade-Off

Figure 12 presents a scatter plot where each method's trade-off variations are shown along two axes: Accuracy (Y-axis) and Δ DP (X-axis). Each point represents a model trained with a different trade-off parameter. Points closer to the top-left are preferred (high accuracy, low ΔDP)

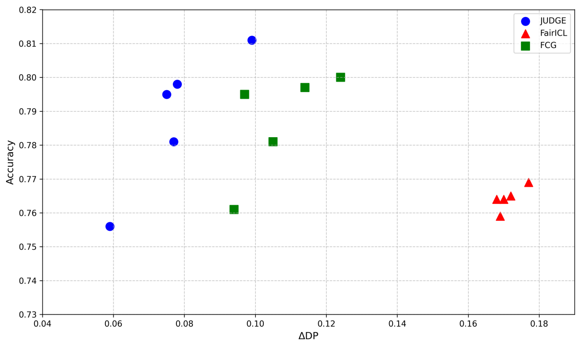


Figure 12: Scatter plot of Accuracy vs. Demographic Parity (ΔDP) for different trade-off parameter settings in JUDGE, FCG, and FairICL.

We observe clear trade-offs for JUDGE and FCG, where higher accuracy comes at the cost of fairness and vice versa, guided by the weighting provided by ω and α. On FairICL, we find the relationship to be less strong, with most points clustered around a similar area.

B.11 Data Splits and Hyperparameters

For all datasets except Adult, we employ a consistent data splitting strategy:

- 20% for test set ($\mathcal{D}_{\text{test}}$)
- 70% for training set ($\mathcal{D}_{\text{train}}$)
- 10% for validation set ($\mathcal{D}_{\text{validation}}$)

For the Adult dataset, which provides a predefined train-test split, we maintain the original test set and split the training set into $\mathcal{D}_{\text{train}}$ (90%) and $\mathcal{D}_{\text{validation}}$ (10%).

It is important to note that this validation set is distinct from the jury set ($\mathcal{J}$) used in our method.

Table 6: Results for COMPAS with 10 demonstrations. Each cell shows $Mean_{S.D.}$

Model	Method	Acc.↑	ΔDP↓	ΔEO↓	MI↓
LLAMA-3-8B	Random	$0.603_{0.013}$	$0.224_{0.010}$	$0.221_{0.009}$	$0.025_{0.002}$
	Balanced	$0.605_{0.007}$	$0.257_{0.006}$	$0.281_{0.006}$	$0.034_{0.002}$
	Cfact.	$0.577_{0.007}$	$0.202_{0.005}$	$0.194_{0.005}$	$0.019_{0.001}$
	Instruct	$0.556_{0.009}$	$0.130_{0.007}$	$0.156_{0.007}$	$0.016_{0.001}$
	FairICL	$0.609_{0.008}$	$0.209_{0.007}$	$0.213_{0.008}$	$0.025_{0.002}$
	FCG	$\mathbf{0.621}_{0.006}$	$0.227_{0.004}$	$0.237_{0.004}$	$0.024_{0.002}$
	FADS	$0.584_{0.006}$	$0.133_{0.008}$	$0.128_{0.004}$	$0.009_{0.001}$
	JUDGE	$0.618_{0.011}$	$\mathbf{0.102}_{0.009}$	$\mathbf{0.114}_{0.009}$	$\mathbf{0.006}_{0.001}$
MISTRAL-7B	Random	$0.527_{0.011}$	$0.130_{0.005}$	$0.157_{0.006}$	$0.019_{0.001}$
	Balanced	$0.517_{0.006}$	$0.089_{0.005}$	$0.115_{0.005}$	$0.017_{0.002}$
	Cfact.	$0.495_{0.009}$	$0.127_{0.007}$	$0.149_{0.007}$	$0.020_{0.004}$
	Instruct	$0.503_{0.011}$	$0.125_{0.008}$	$0.141_{0.009}$	$0.017_{0.002}$
	FairICL	$0.514_{0.006}$	$0.110_{0.003}$	$0.127_{0.004}$	$0.018_{0.006}$
	FCG	$\mathbf{0.547}_{0.011}$	$0.153_{0.007}$	$0.129_{0.008}$	$0.015_{0.003}$
	FADS	$0.536_{0.014}$	$0.129_{0.008}$	$0.137_{0.017}$	$0.018_{0.003}$
	JUDGE	$0.538_{0.008}$	$\mathbf{0.056}_{0.004}$	$\mathbf{0.055}_{0.004}$	$\mathbf{0.007}_{0.001}$
GEMMA-2-9B	Random	$0.610_{0.006}$	$0.311_{0.005}$	$0.298_{0.006}$	$0.048_{0.002}$
	Balanced	$0.624_{0.007}$	$0.324_{0.005}$	$0.303_{0.005}$	$0.054_{0.005}$
	Cfact.	$0.597_{0.009}$	$0.255_{0.008}$	$0.248_{0.008}$	$0.039_{0.003}$
	Instruct	$0.608_{0.012}$	$0.292_{0.013}$	$0.301_{0.011}$	$0.046_{0.005}$
	FairICL	$0.631_{0.009}$	$0.272_{0.009}$	$0.281_{0.007}$	$0.044_{0.005}$
	FCG	$0.645_{0.006}$	$0.119_{0.004}$	$0.128_{0.006}$	$0.009_{0.002}$
	FADS	$0.628_{0.009}$	$0.289_{0.007}$	$0.277_{0.014}$	$0.041_{0.003}$
	JUDGE	$\mathbf{0.648}_{0.006}$	$\mathbf{0.059}_{0.003}$	$\mathbf{0.035}_{0.000}$	$\mathbf{0.002}_{0.000}$
QWEN-2.5-32B	Random	$0.641_{0.006}$	$0.231_{0.004}$	$0.210_{0.004}$	$0.024_{0.002}$
	Balanced	$0.658_{0.009}$	$0.229_{0.011}$	$0.238_{0.010}$	$0.029_{0.005}$
	Cfact.	$0.653_{0.009}$	$0.197_{0.005}$	$0.187_{0.007}$	$0.020_{0.003}$
	Instruct	$0.644_{0.010}$	$0.213_{0.007}$	$0.188_{0.007}$	$0.023_{0.005}$
	FairICL	$0.642_{0.009}$	$0.202_{0.006}$	$0.211_{0.007}$	$0.023_{0.002}$
	FCG	$0.631_{0.008}$	$0.167_{0.004}$	$0.191_{0.005}$	$0.021_{0.003}$
	FADS	$\mathbf{0.659}_{0.012}$	$0.199_{0.011}$	$0.170_{0.014}$	$0.020_{0.003}$
	JUDGE	$0.652_{0.006}$	$\mathbf{0.111}_{0.004}$	$\mathbf{0.129}_{0.003}$	$\mathbf{0.011}_{0.002}$

Table 7: Results for ACS with 10 demonstrations. Each cell shows $Mean_{S.D.}$

Model	Method	Acc.↑	ΔDP↓	ΔEO↓	MI↓
LLAMA-3-8B	Random	$0.699_{0.011}$	$0.108_{0.007}$	$0.091_{0.008}$	$0.006_{0.001}$
	Balanced	$0.695_{0.010}$	$0.095_{0.007}$	$0.088_{0.006}$	$0.004_{0.000}$
	Cfact.	$0.682_{0.011}$	$0.090_{0.004}$	$0.091_{0.003}$	$0.004_{0.001}$
	Instruct	$0.693_{0.013}$	$0.103_{0.008}$	$0.099_{0.009}$	$0.008_{0.001}$
	FairICL	$0.692_{0.009}$	$0.089_{0.004}$	$0.098_{0.005}$	$0.005_{0.002}$
	FCG	$0.755_{0.007}$	$0.059_{0.003}$	$\mathbf{0.056}_{0.008}$	$0.002_{0.006}$
	FADS	$0.723_{0.010}$	$0.121_{0.006}$	$0.114_{0.008}$	$0.007_{0.001}$
	JUDGE	$\mathbf{0.766}_{0.009}$	$\mathbf{0.024}_{0.004}$	$0.059_{0.006}$	$\mathbf{0.001}_{0.000}$
MISTRAL-7B	Random	$0.648_{0.08}$	$0.085_{0.004}$	$0.042_{0.008}$	$0.004_{0.001}$
	Balanced	$0.571_{0.011}$	$0.061_{0.010}$	$0.034_{0.004}$	$0.003_{0.000}$
	Cfact.	$0.612_{0.07}$	$0.077_{0.003}$	$0.058_{0.004}$	$0.004_{0.001}$
	Instruct	$0.607_{0.011}$	$0.092_{0.009}$	$0.099_{0.007}$	$0.006_{0.001}$
	FairICL	$0.622_{0.008}$	$0.081_{0.005}$	$0.057_{0.004}$	$0.005_{0.001}$
	FCG	$0.650_{0.006}$	$0.048_{0.004}$	$0.067_{0.003}$	$0.002_{0.000}$
	FADS	$0.636_{0.014}$	$0.080_{0.006}$	$\mathbf{0.026}_{0.004}$	$0.004_{0.000}$
	JUDGE	$\mathbf{0.655}_{0.009}$	$\mathbf{0.029}_{0.004}$	$0.037_{0.002}$	$\mathbf{0.001}_{0.000}$
GEMMA-2-9B	Random	$0.712_{0.012}$	$0.218_{0.008}$	$0.262_{0.009}$	$0.029_{0.003}$
	Balanced	$0.713_{0.012}$	$0.201_{0.008}$	$0.223_{0.006}$	$0.027_{0.002}$
	Cfact.	$0.719_{0.008}$	$0.206_{0.007}$	$0.238_{0.008}$	$0.025_{0.002}$
	Instruct	$0.707_{0.015}$	$0.231_{0.009}$	$0.281_{0.011}$	$0.038_{0.004}$
	FairICL	$0.718_{0.021}$	$0.208_{0.008}$	$0.224_{0.007}$	$0.023_{0.004}$
	FCG	$0.715_{0.009}$	$0.125_{0.006}$	$0.129_{0.008}$	$0.016_{0.002}$
	FADS	$\mathbf{0.725}_{0.009}$	$0.217_{0.011}$	$0.264_{0.009}$	$0.032_{0.004}$
	JUDGE	$0.722_{0.011}$	$\mathbf{0.113}_{0.004}$	$\mathbf{0.118}_{0.005}$	$\mathbf{0.013}_{0.001}$
QWEN-2.5-32B	Random	$0.736_{0.013}$	$0.111_{0.009}$	$0.046_{0.009}$	$0.006_{0.001}$
	Balanced	$0.730_{0.008}$	$0.096_{0.008}$	$\mathbf{0.019}_{0.003}$	$0.003_{0.000}$
	Cfact.	$0.737_{0.009}$	$0.091_{0.004}$	$0.048_{0.002}$	$0.005_{0.001}$
	Instruct	$0.741_{0.011}$	$0.181_{0.006}$	$0.105_{0.007}$	$0.016_{0.002}$
	FairICL	$0.733_{0.012}$	$0.089_{0.006}$	$0.094_{0.009}$	$0.004_{0.001}$
	FCG	$0.731_{0.007}$	$0.037_{0.005}$	$0.044_{0.005}$	$0.002_{0.000}$
	FADS	$\mathbf{0.751}_{0.004}$	$0.122_{0.004}$	$0.045_{0.003}$	$0.006_{0.000}$
	JUDGE	$0.740_{0.011}$	$\mathbf{0.028}_{0.004}$	$0.039_{0.006}$	$\mathbf{0.001}_{0.001}$

While the jury set is constructed from the training data to guide demonstration selection and is typically very small, the validation set is used exclusively for hyperparameter tuning.

To tune hyperparameters, we conduct a systematic grid search over two key hyperparameters:

1. The fairness-accuracy trade-off parameter ω in the range $[0.3, 0.9]$ with steps of 0.1
2. The number of examples per group-label combination m in the jury set, testing values $\{15, 20, 25, 50\}$

In Section 5.7, we demonstrated that jury sizes beyond $m = 50$ yield diminishing returns, while very small values ($m \in \{1, 2, 3, 5, 10\}$) show substantial performance gaps in fairness and accuracy. Based on these observations, we focus our parameter search on the more practical intermediate range. For jury set size selection, we do as follows: Starting from smaller values, we incrementally evaluate larger jury sizes until we observe diminishing returns in performance on the validation set. Specifically, if the relative improvement in both accuracy and fairness metrics between two consecutive jury sizes falls below 1%, we stop increasing the size. This process led to the selection of $m = 25$ for Adult and COMPAS datasets, and $m = 50$ for Law School and ACS datasets.

The larger jury sizes for Law School and ACS datasets were chosen because these datasets exhibited continued performance improvements with larger jury sizes. In contrast, Adult and COMPAS datasets showed performance saturation at $m = 25$, making larger jury sizes unnecessary.

For the fairness-accuracy trade-off parameter ω, we select the value that achieves the lowest ΔDP on $\mathcal{D}_{\text{validation}}$ while maintaining accuracy within 3% of the best performing configuration.

For the reduced candidate set size $|\mathcal{D}_{\text{reduced}}|$, we empirically evaluated different percentages of $\mathcal{D}_{\text{candidates}}$ from 1% to 5% in steps of 1%. When

Table 8: Results for Law School with 10 demonstrations. Each cell shows $Mean_{S.D.}$

	Method	Acc.↑	ΔDP↓	ΔEO↓	MI↓
LLAMA-3-8B	Random	$0.913_{0.021}$	$0.193_{0.013}$	$0.353_{0.015}$	$0.036_{0.005}$
	Balanced	$0.688_{0.020}$	$0.388_{0.014}$	$0.357_{0.016}$	$0.044_{0.005}$
	Cfact.	$0.912_{0.021}$	$0.220_{0.017}$	$0.475_{0.021}$	$0.039_{0.004}$
	Instruct	$0.905_{0.022}$	$0.177_{0.017}$	$0.321_{0.020}$	$0.017_{0.002}$
	FairICL	$0.903_{0.013}$	$0.331_{0.009}$	$0.328_{0.007}$	$0.030_{0.004}$
	FCG	$\mathbf{0.932}_{0.011}$	$0.076_{0.006}$	$0.239_{0.011}$	$0.019_{0.003}$
	FADS	$0.889_{0.005}$	$0.241_{0.003}$	$0.453_{0.003}$	$0.035_{0.002}$
	JUDGE	$0.922_{0.0122}$	$\mathbf{0.069}_{0.005}$	$\mathbf{0.166}_{0.006}$	$\mathbf{0.012}_{0.001}$
MISTRAL-7B	Random	$0.924_{0.015}$	$0.174_{0.009}$	$0.287_{0.011}$	$0.025_{0.002}$
	Balanced	$0.899_{0.009}$	$0.224_{0.008}$	$0.438_{0.006}$	$0.034_{0.003}$
	Cfact.	$0.919_{0.018}$	$0.194_{0.008}$	$0.402_{0.013}$	$0.027_{0.003}$
	Instruct	$0.933_{0.012}$	$0.031_{0.006}$	$\mathbf{0.039}_{0.003}$	$\mathbf{0.001}_{0.000}$
	FairICL	$0.927_{0.017}$	$0.179_{0.004}$	$0.268_{0.009}$	$0.024_{0.002}$
	FCG	$0.941_{0.022}$	$0.048_{0.004}$	$0.081_{0.005}$	$0.012_{0.006}$
	FADS	$0.934_{0.007}$	$0.108_{0.006}$	$0.204_{0.006}$	$0.021_{0.003}$
	JUDGE	$\mathbf{0.949}_{0.019}$	$\mathbf{0.026}_{0.004}$	$0.061_{0.009}$	$0.004_{0.000}$
GEMMA-2-9B	Random	$0.876_{0.006}$	$0.331_{0.005}$	$0.439_{0.003}$	$0.056_{0.003}$
	Balanced	$0.745_{0.012}$	$0.416_{0.009}$	$0.376_{0.006}$	$0.053_{0.004}$
	Cfact.	$0.791_{0.009}$	$0.371_{0.005}$	$0.340_{0.006}$	$0.047_{0.004}$
	Instruct	$0.881_{0.005}$	$0.362_{0.004}$	$0.534_{0.007}$	$0.053_{0.001}$
	FairICL	$0.862_{0.013}$	$0.314_{0.008}$	$0.338_{0.007}$	$0.043_{0.002}$
	FCG	$0.858_{0.013}$	$0.229_{0.008}$	$0.254_{0.009}$	$0.031_{0.002}$
	FADS	$\mathbf{0.881}_{0.006}$	$0.290_{0.006}$	$0.579_{0.011}$	$0.054_{0.004}$
	JUDGE	$0.862_{0.010}$	$\mathbf{0.212}_{0.004}$	$\mathbf{0.199}_{0.005}$	$\mathbf{0.021}_{0.001}$
QWEN-2.5-32B	Random	$0.882_{0.005}$	$0.295_{0.007}$	$0.513_{0.007}$	$0.048_{0.003}$
	Balanced	$0.842_{0.008}$	$0.316_{0.005}$	$0.554_{0.012}$	$0.057_{0.005}$
	Cfact.	$0.845_{0.006}$	$0.394_{0.005}$	$0.541_{0.004}$	$0.064_{0.002}$
	Instruct	$\mathbf{0.897}_{0.014}$	$0.351_{0.007}$	$0.628_{0.008}$	$0.079_{0.003}$
	FairICL	$0.878_{0.015}$	$0.281_{0.009}$	$0.524_{0.005}$	$0.053_{0.003}$
	FCG	$0.879_{0.017}$	$0.252_{0.013}$	$0.318_{0.016}$	$0.036_{0.006}$
	FADS	$0.896_{0.022}$	$0.230_{0.011}$	$0.520_{0.015}$	$0.045_{0.009}$
	JUDGE	$0.883_{0.021}$	$\mathbf{0.203}_{0.012}$	$\mathbf{0.288}_{0.017}$	$\mathbf{0.028}_{0.001}$

increasing the size from 1% to 3%, we observed average improvements of 2-3% in both accuracy and fairness metrics across all datasets. However, further increases beyond 3% showed minimal gains ($< 0.5\%$ improvement) while significantly increasing computational overhead. Therefore, we set $|\mathcal{D}_{\text{reduced}}|$ to 3% of $|\mathcal{D}_{\text{candidates}}|$ for all experiments.

All hyperparameter tuning is performed using only the validation set, with the test set remaining completely held out until final evaluation. To summarize, in our extensive experiments, we find that setting $|\mathcal{D}_{\text{reduced}}|$ to 3% of $|\mathcal{D}_{\text{candidates}}|$ provides consistently good results across different datasets and models. Further, while $m = 50$ examples per group-label combination works reliably across all settings, $m = 25$ is often sufficient and more computationally efficient. Values of ω above 0.5, particularly around 0.7, tend to provide better fairness-accuracy trade-offs. We intentionally keep the granularity of these parameter searches relatively coarse to maintain computational efficiency while still achieving strong performance. These settings can serve as reliable defaults for practitioners.

B.12 Models and Software Used

Experiments were conducted using PyTorch (2.4.1), and all models we use are publicly available on HuggingFace. For SentenceBERT, we use the SentenceTransformers package, and we specifically use the "all-mpnet-base-v2" variant, which has the best reported performance. For the LLMs, we use the base variants of all models (LLAMA-3-8B, MISTRAL-7B, QWEN-2.5-32B, and Gemma-2-9B). We downloaded them from HuggingFace via the Transformers library, and we note that some of them are gated models that require access tokens. For inference on these models, we turn off sampling in all experiments, to get the desired deterministic behavior for In-Context Learning.

B.13 Computing Infrastructure

The experiments in this paper were conducted across three different computing environments. System A consisted of an Intel(R) Xeon(R) CPU E5-2680 v4 @ 2.40GHz processor with 512GB RAM and 8 NVIDIA V100 GPUs. System B utilized an AMD Ryzen Threadripper PRO 5955WX (16 cores) with 256GB RAM and dual NVIDIA RTX 3090 GPUs. System C provided limited access to a high-performance computing cluster equipped with dual 64-core AMD EPYC 7763 processors, 256GB DDR4 memory, and 4 NVIDIA A100 GPUs. While we did not formally track GPU hours, we estimate that the total computational effort across all experiments, including baseline implementations, LLM training and inference, methodology development, and ablation studies exceeded well over a thousand GPU hours. This estimate encompasses the entire research and development cycle, including exploratory experiments, hyperparameter optimization, model training iterations, and evaluation runs.

C Complexity Comparison Across Methods

Here, we provide a detailed comparison of the computational complexity of various demonstration selection methods in terms of **LLM calls**.

While all methods involve inference over the test set which uses LLM calls, meaning they inherently contain an $O(|D_{\text{test}}|)$ term, this is dominated by larger computational factors in all but the Naïve baselines, and is therefore omitted from the complexity expressions for clarity for the other baselines.

C.1 Naïve Baselines (Counterfactual, Instruct, Random, etc.)

These methods do not optimize demonstrations based on LLM feedback, meaning the only LLM calls occur during test-time inference:

$$O(|D_{\text{test}}|)$$

C.2 FADS (Fairness-Aware Demonstration Selection)

The primary computational cost in FADS arises from the **model bias mitigation step**, where LLM queries are made for all samples within a subset of clusters retained after filtering for data-bias.

FADS first partitions the training data D_{train} into K clusters using K-means. After clustering, only N_d clusters are retained for fairness-aware demonstration selection. Since each cluster contains approximately $|D_{\text{train}}|/K$ samples, the total number of LLM queries in this step is:

$$O(N_d \cdot |D_{\text{train}}|/K)$$

where:

- N_d is the number of clusters retained after filtering.
- $|D_{\text{train}}|$ is the total size of the training dataset.
- K is the number of clusters initially created.

After this filtering step, demonstrations are selected dynamically for each test instance based on semantic similarity, but this retrieval step is lightweight and does not require LLM calls. Thus, the final complexity of FADS in terms of LLM calls is:

$$O(N_d \cdot |D_{\text{train}}|/K)$$

C.3 FCG (Fairness via Clustering-Genetic Algorithm)

FCG iteratively refines demonstration selection using a **genetic algorithm**, making multiple LLM calls per validation sample over I iterations:

$$O(I \cdot |D_{\text{dev}}| \cdot S)$$

where S is the number of subgroups as defined in the paper, and D_{dev} is the validation dataset used to assess demonstration fairness.

C.4 FairICL (Fair In-Context Learning via Latent Concept Variables)

FairICL requires additional LLM calls for **latent concept learning**, followed by likelihood-based demonstration selection:

$$O\left(T \cdot \frac{|D_{\text{train}}|}{B}\right) + O(|D_{\text{train}}|)$$

where T is the number of training epochs, B is batch size, and D_{train} is the training dataset used to learn the latent concept variable.

C.5 Comparison Summary

Table 9 summarizes the computational complexity of various demonstration selection methods in terms of **LLM calls**, which dominate the overall compute cost.

Simpler baselines, such as **Balanced, Random, Counterfactual, and Instruct** require only $O(|D_{\text{test}}|)$ LLM calls, making them the most computationally efficient but very often lead to sub-optimal in fairness and accuracy as they do not optimize the demonstration specifically based on the LLM's feedback.

FADS significantly reduces LLM calls by leveraging **clustering and heuristic fairness scoring** before querying the LLM. Its complexity, $O(N_d \cdot |D_{\text{train}}|/K)$, is linear in the training set size but avoids expensive iterative selection.

FairICL introduces an additional concept-learning step that requires learning a **latent fairness representation**. This step adds overhead, making its complexity $O(T \cdot \frac{|D_{\text{train}}|}{B}) + O(|D_{\text{train}}|)$, where T and B are training epochs and batch size, respectively. This method offers improved fairness guarantees at the cost of increased compute.

FCG employs a **genetic algorithm** that iteratively refines demonstration selection using validation data. This results in $O(I \cdot |D_{\text{dev}}| \cdot S)$ complexity, where I is the number of iterations and S is the number of demographic subgroups considered. The actual computational cost of FCG depends on the choice of these parameters. When $|D_{\text{dev}}|$ is large or I is high, FCG can be computationally expensive, whereas for smaller values, it may be comparable to or even more efficient than methods that process larger training subsets.

Exhaustive search, which evaluates all possible subsets of K-shot demonstrations, is prohibitively expensive with complexity $O(N^K)$, making it infeasible for large N and K, as described in Section 4.

Method	LLM Calls Complexity
Naïve Methods (Random, Counterfactual, Instruct, etc.)	$O(\lvert D_{\text{test}} \rvert)$
FADS (Fairness-Aware Demonstration Selection)	$O(N_d \cdot \lvert D_{\text{train}} \rvert / K)$
FairICL (Latent Concept Learning)	$O(T \cdot \frac{\lvert D_{\text{train}} \rvert}{B}) + O(\lvert D_{\text{train}} \rvert)$
FCG (Clustering-Genetic Algorithm)	$O(I \cdot \lvert D_{\text{dev}} \rvert \cdot S)$
Exhaustive Search (Global Optimal Set)	$O(N^K)$
JUDGE (Ours)	$O(k \cdot \lvert \mathcal{D}_{\text{reduced}} \rvert \cdot \lvert \mathcal{J} \rvert)$

Table 9: Comparison of LLM Calls Complexity Across Different Methods

JUDGE constructs **a single optimized demonstration set**. Its complexity, $O(k \cdot |\mathcal{D}_{\text{reduced}}| \cdot |\mathcal{J}|)$, scales with the reduced candidate set size $|\mathcal{D}_{\text{reduced}}|$, the number of fairness evaluations $|\mathcal{J}|$, as described in Section 4.

Overall, aside from the simpler baselines which do not utilize any feedback from the LLM, the computational efficiency of these methods depends on the specific parameter choices. For example, the relative efficiency of JUDGE compared to FCG depends on how $|\mathcal{D}_{\text{reduced}}|$ compares to $|D_{\text{dev}}|$ and how the parameters I in FCG and $\mathcal{J}$ in JUDGE are set.