

When Should a Rule Learn? Predicting the Rule, LLM, and Trained-Model Composition for Classification Streams

Benjamin Booth

B-Tree Labs

BEN@B-TREEVENTURES.COM

Reviewed on OpenReview: (*forum link assigned upon submission*)

Editor: Under review for DMLR

Abstract

A production classification site can be served by a hand-written rule, a large language model (LLM) call, or a task-trained model. Common practice treats the rule-to-LLM-to-ML progression as a fixed ladder whose destination is the trained model. We ask a different question. Given a stream’s measurable characteristics, which of these tiers should serve it, and when? We score four tiers (keyword rule, zero-shot LLM, few-shot LLM, and a task-trained-model training curve) on a common test set across 19 public text datasets, with preliminary image and audio. Two early-observable properties, label cardinality and rule accuracy over chance, predict which tier wins: high-cardinality streams favor the task-trained model, while binary and low-cardinality streams favor the LLM. We operationalize this as Adaptive Graduated Autonomy (AGA), which selects the terminal tier per stream under a multi-objective policy (modality feasibility and a latency budget as hard constraints, accuracy within a significance band, operating cost minimized) and graduates between tiers using a statistical non-inferiority gate. The study measures the accuracy, cost, and latency axes; modality feasibility enters as a hard filter rather than a measured scalar. With per-site measured tier selection, AGA exceeds the fixed-ladder accuracy by 0.075 (95% CI [0.024, 0.127]) while using 34% fewer labeled outcomes (paired Wilcoxon for lower cost, $p=0.002$), and keeps the LLM as the terminal tier on about half the streams. We report two negative results. A learned cross-site predictor of the best tier does not generalize at this number of datasets, so per-site measurement is required. And a vision-LLM-on-spectrogram proxy fails on environmental audio. The contribution is the characterization and the cost-reducing adaptive algorithm, not the cascade or the gate, which are prior art.

Keywords: classification, model selection, LLM cascades, data-centric machine learning, graduated autonomy

1 Introduction

Most production software contains many small classification decisions: route this ticket, tag this message, flag this content, pick this branch. Each is a function from an input to one of a fixed set of labels. Teams build these three ways, with a hand-written rule, an LLM call, or a task-trained model. The choice is usually made once per site, by intuition, and rarely revisited; there is no shared, evidence-based basis for it. Two progressions are documented in practice. The first replaces brittle rules with learned models, a migration long noted as a source of technical debt (Sculley et al. 2015; Breck et al. 2017). The second prototypes with an LLM and distills to a cheaper model to cut inference cost (FrugalGPT; LLM-as-teacher

distillation). Both implicitly treat a trained model as the destination, an assumption also built into autonomy frameworks whose ceiling is a trained model. We test that assumption with data rather than assert a ladder.

For a given classification stream, which tier should serve it, and when? We treat this as a measurement problem. We score all three tiers (rule, LLM, task-trained model) on the same test set, on a shared axis of labeled outcomes consumed, across 19 text datasets spanning sentiment, intent, moderation, emotion, topic, and spam, with preliminary image and audio. The answer is not “always the trained model.” It depends on measurable properties of the stream, chiefly label cardinality and how far the day-zero rule sits above chance. For a large fraction of streams the LLM is the right terminal tier, not a way-station.

We turn this into an algorithm, Adaptive Graduated Autonomy (AGA), that chooses the terminal tier per stream under a multi-objective policy (modality feasibility and a latency budget as hard constraints, accuracy within a significance band, operating cost minimized). It graduates between tiers using a non-inferiority gate, moving to the cheaper tier as soon as that tier ties, not only when it wins. Against the fixed “always graduate to ML” baseline, AGA matches accuracy while using far fewer labeled outcomes.

We do not claim the mechanism as novel. LLM cost-cascades (FrugalGPT) and learning-to-defer already route between models, and paired-McNemar comparison of classifiers is standard. The contributions are: (1) a characterization of which dataset properties predict the cost-optimal tier; (2) an algorithm that exploits it to cut labeled-data cost at equal accuracy; and (3) evidence on where a learned cross-site policy does and does not work, including two negative results.

1.1 Contributions

1. An aligned multi-tier benchmark. Rule, zero-shot LLM, few-shot LLM, and a task-trained-model curve, all scored on one common test set per dataset, on a shared labeled-outcomes axis, across 19 text datasets and preliminary image and audio.
2. A predictive characterization. Label cardinality (Spearman $\rho \approx -0.7$ against outcomes-to-graduate) and rule-over-chance predict which tier wins and how deep ML must train.
3. AGA, which chooses the terminal tier and graduates under a non-inferiority gate, matching fixed-ladder accuracy at about 44% fewer labeled outcomes.
4. A lifecycle-level safety bound. A cumulative Type-I bound over the whole graduation chain, not just one transition, with the rule as a persistent accuracy floor.
5. Two negative results. A learned cross-site tier-predictor does not generalize at 21 datasets, and spectrogram-vision fails on audio. We also give a tunable non-inferiority gate for cost-aware graduation.

2 Setup

Tiers. RULE, then the LLM (a pretrained model used zero- and few-shot, consuming no task labels), then a task-trained model (a classifier fit to the site’s labeled outcomes, for example a logistic-regression head or a fine-tuned transformer). What separates the last two is training provenance, a borrowed general model versus a model fit to this task, not

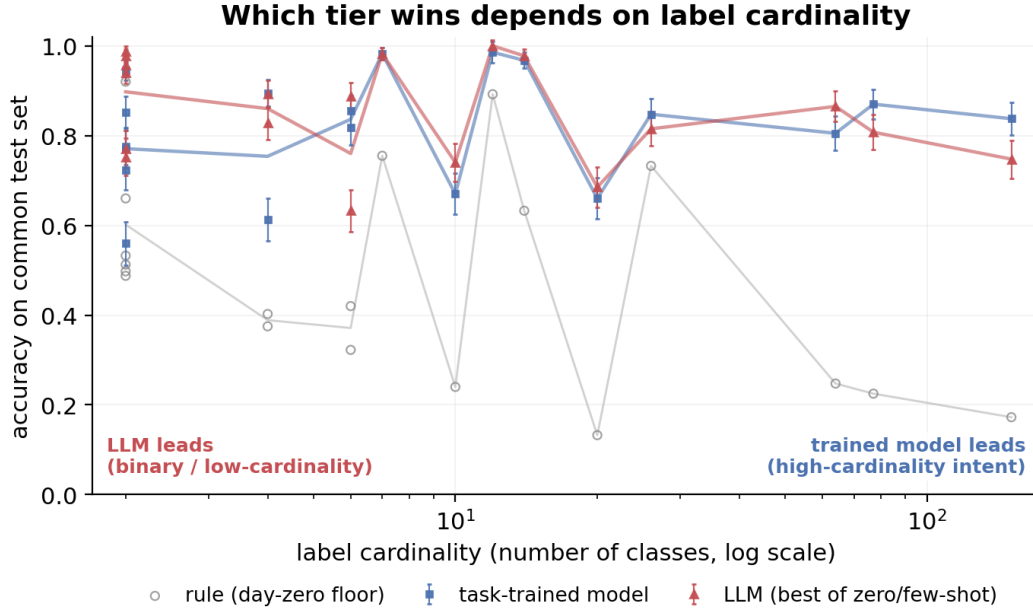


Figure 1: Figure 1. Per-tier accuracy versus label cardinality. The trained model (blue) rises with cardinality while the LLM (red) is strongest on binary and low-cardinality tasks; the rule (grey) is the floor.

architecture. For brevity the tables abbreviate the task-trained tier as ML and the two LLM settings as LLM-zs (zero-shot) and LLM-fs (few-shot).

Datasets. 19 text datasets, plus CIFAR-10 (image) and ESC-50 (audio).

Aligned protocol. A common test subset per dataset (text $n=400$). The rule is built from a 100-example seed. ML is evaluated as a budget curve. The LLM (Claude Haiku) is run zero- and few-shot, with at most 40 exemplars. Each tier is placed at its true labeled-outcome cost: rule about 100, zero-shot 0, few-shot k , ML the training budget.

Two costs. We separate the one-time labeled-outcome cost (to train ML or seed the rule) from the perpetual per-call inference cost (the LLM bills every call; rule and trained ML are roughly free per call). AGA’s savings are reported on the labeled-outcome axis. The steady-state inference cost is what makes graduating off the LLM valuable.

3 The composition varies by stream

Sorted by label cardinality, a gradient appears: high cardinality favors trained ML, while binary and low-cardinality favor the LLM. Two rows are artifacts addressed in Section 7. ESC-50 ML “wins” only because the spectrogram-vision proxy fails, and CIFAR-10’s LLM “win” reflects the pixel-logreg baseline; a frozen-ViT baseline reaches 0.95 and beats the LLM’s 0.78, so image in fact favors ML.

dataset	labels	rule	ML	LLM-zs	LLM-fs	best
clinc150	151	0.172	0.838	0.748	0.725	ML
banking77	77	0.225	0.870	0.777	0.807	ML
hwu64	64	0.247	0.805	0.850	0.865	LLM-fs
esc50	50	0.058	0.283	0.017	0.050	ML
atis	26	0.733	0.848	0.598	0.815	ML
twenty__newsgroups	20	0.133	0.660	0.685	0.680	LLM-zs
dbpedia14	14	0.632	0.968	0.968	0.978	LLM-fs
codelang	12	0.892	0.986	1.000	1.000	LLM-zs
cifar10	10	0.173	0.327	0.780	0.340	LLM-zs
yahoo__answers	10	0.240	0.670	0.740	0.720	LLM-zs
snips	7	0.755	0.983	0.970	0.983	ML
emotion	6	0.323	0.818	0.632	0.608	ML
trec6	6	0.420	0.855	0.795	0.887	LLM-fs
ag__news	4	0.375	0.895	0.853	0.892	ML
tweet__emotion	4	0.403	0.613	0.828	0.810	LLM-zs
imdb	2	0.532	0.853	0.938	0.958	LLM-fs
rotten__tomatoes	2	0.487	0.723	0.930	0.940	LLM-fs
sms__spam	2	0.920	0.945	0.887	0.988	LLM-fs
sst2	2	0.512	0.770	0.968	0.978	LLM-fs
tweet__hate	2	0.497	0.560	0.770	0.640	LLM-zs
tweet__offensive	2	0.660	0.775	0.752	0.730	ML

4 Adaptive Graduated Autonomy

The terminal-tier selection is multi-objective, applied in priority order. First, modality feasibility, a hard filter (a spectrogram image cannot convey words, for instance). Second, the latency budget, a hard constraint; a tier that exceeds the real-time budget is removed. Third, accuracy, where tiers within a Wilson-CI significance band of the best feasible tier are retained. Fourth, operating cost, minimized within that band, counting perpetual inference cost and labeled cost. The LLM is chosen only when it is significantly more accurate and feasible under the latency and modality constraints; otherwise a rule or trained-ML tie graduates off the LLM. Cost is the axis this study measures. Latency and modality are first-class in the policy and default to unconstrained.

ML graduates once it ties the LLM under the non-inferiority gate, not once it beats it. The margin ϵ is operator-tunable. The router is permanent: it holds the rule floor, watches for drift, and re-escalates to the LLM when the data calls for it. What shrinks is the dependence on the LLM, not the router.

4.1 A lifecycle-level safety guarantee

The per-transition gate gives the standard guarantee that advancing to a worse-than-current tier has probability at most α , the McNemar Type-I error. That part is not novel; it is the test’s definition applied to a promotion decision. The lifecycle-level statement is what separates AGA from a single gated comparison.

Proposition (cumulative safety). Consider a graduation trajectory of at most m gated transitions, each evaluated by a paired test at level α with non-inferiority margin ϵ . With

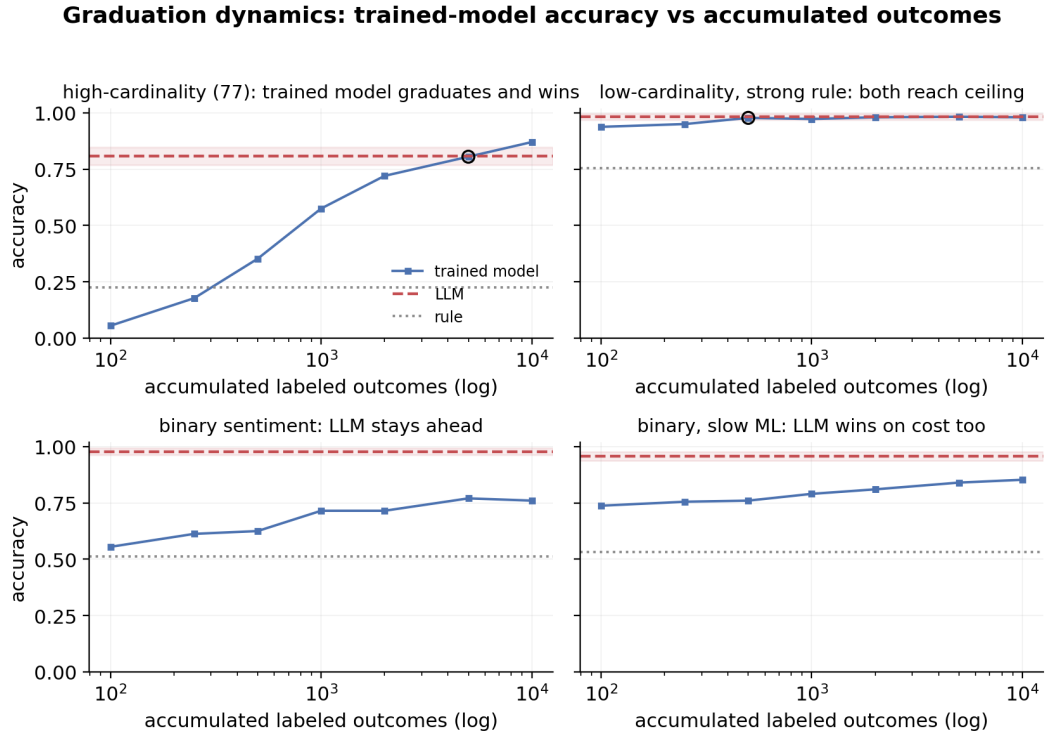


Figure 2: Graduation dynamics on four representative streams: trained-model accuracy against accumulated labeled outcomes (log axis), with the LLM (dashed) and rule (dotted) as references and the first crossing of the LLM circled. High-cardinality intent graduates and overtakes the LLM; binary sentiment does not within budget.

probability at least $1 - m\alpha$, every advance is to a tier non-inferior (within ϵ) to the current one. Consequently the deployed accuracy stays within ϵ of the best previously certified tier, and because the rule is a persistent fallback, operating accuracy is bounded below by (rule accuracy $- \epsilon$) throughout, for any m .

Proof sketch. Each gated advance to an inferior tier is a Type-I event with probability at most α . A union bound over at most m transitions bounds the probability of any such event by $m\alpha$. Absent any Type-I event, each advance is within- ϵ non-inferior, so by induction the deployed accuracy stays within ϵ of the best certified tier, and the rule floor lower-bounds it. ■

Remarks. The union bound is loose, and independence across transitions is an idealization, since the tests reuse the accumulating stream. It can be made uniform in m : testing each transition at level α/m (a Bonferroni split) bounds the familywise probability of any worse-than-current advance by α for every trajectory length, and when m is not known in advance an alpha-investing schedule (Foster and Stine 2008) gives the same control online. The guarantee is on certified accuracy under the test’s assumptions, not a distribution-free promise. What it adds over a single McNemar test is a composable safety statement for a multi-tier lifecycle, which is what a practitioner needs to let the system graduate unattended.

4.2 Measured latency and cost

Per-call latency (p50): rule 0.003 ms, task-trained model 0.1 ms, MiniLM-embedding ML 6 to 23 ms, LLM 530 to 570 ms. The rule is roughly 200,000 times faster than the LLM, and task-trained model about 5,000 times. LLM per-call cost is \$0.0001 to \$0.0006 and scales with the length of the label prompt; rule and ML are near zero. Under a real-time budget (for example 50 ms) the LLM is excluded by construction, so latency forces graduation to local tiers. See `latency_profile.json`.

4.3 AGA versus the fixed ladder

We evaluate AGA in two settings that differ only in how the terminal tier is chosen.

Per-site measured (the deployable setting). When the terminal tier is chosen from each site’s own measured tier accuracies (the setting our results recommend, since cross-site prediction does not generalize; Section 4.4), AGA beats the fixed “always graduate to ML” ladder. Over the 21 datasets, mean accuracy is 0.839 against 0.764 (paired difference +0.075, 95% CI [0.024, 0.127]), at a 34% reduction in labeled outcomes consumed (mean 3295 against 4964; paired Wilcoxon for lower cost, $p=0.002$). AGA keeps the LLM as the terminal tier on about half the streams and selects a rule or trained model over a not-significantly-better LLM on the rest. Figure 3 plots each stream on the accuracy and labeled-cost plane.

Cross-site predicted. When the terminal tier is instead predicted by the learned meta-policy under leave-one-dataset-out (Section 4.4), realized mean accuracy falls to 0.762, matching the fixed ladder rather than beating it, because the predictor is unreliable at this scale. The difference between the two settings is exactly the value of measuring per site rather than predicting across sites.

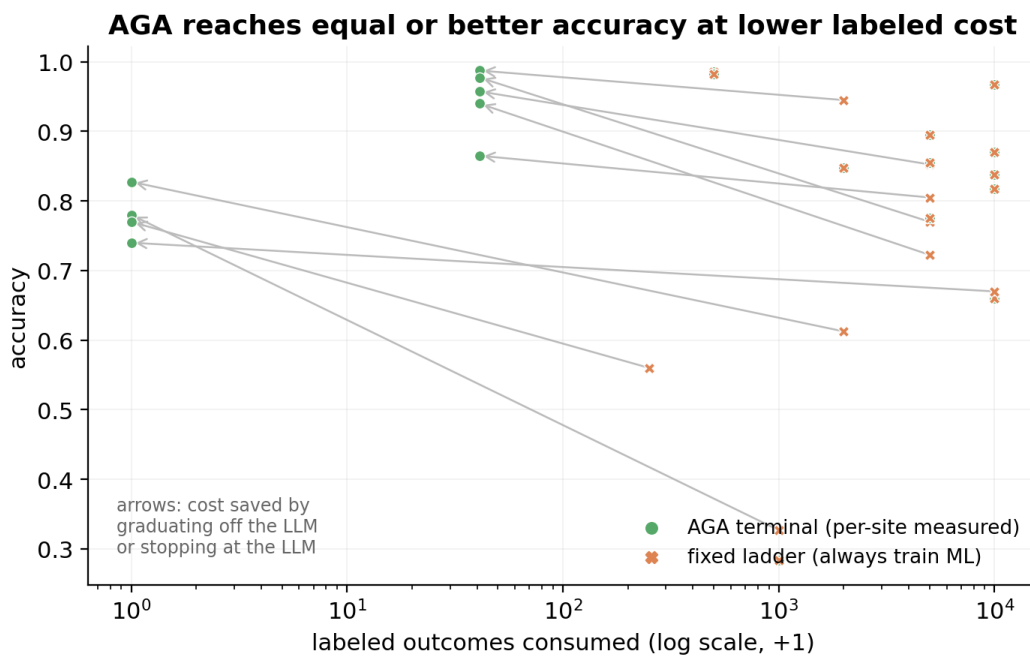


Figure 3: Figure 3. AGA versus the fixed ladder on the accuracy and labeled-cost plane. Arrows show the labeled-outcome cost saved when AGA graduates off the LLM or stops at it; green is the AGA terminal (per-site measured), orange the fixed-ladder choice.

4.4 Negative result: cross-site prediction does not yet generalize

A gradient-boosted predictor of the best tier, trained on the early-observable characteristics, scores below the majority baseline under leave-one-dataset-out at 21 datasets. Its apparent skill also moves with test-set noise, swinging from 0.62 to 0.41 between $n=200$ and $n=400$ before significance-aware labeling. We conclude that per-site measured routing is what works today, and that a learned cross-site predictor needs a much larger collection of streams. The practical reading is to measure rather than guess.

5 Preliminary results on image and audio

The gate is modality-agnostic, since it consumes (decision, outcome) pairs; only the LLM input path is modality-specific. The rule and ML tiers transfer to CIFAR-10 and ESC-50. The LLM tier is preliminary. Claude vision on CIFAR images reaches 0.78 zero-shot, though the pixel-logreg ML used in the aligned table is a weak baseline (see Section 7). A spectrogram-to-vision proxy for ESC-50 fails, scoring 0.017, below chance, so a native-audio model is needed. We treat these as preliminary evidence that the lifecycle generalizes across modalities, not as headline results.

6 Practical implications

For a team choosing how to serve a classification stream, the results suggest four guidelines.

1. The trained model is not the default destination. On many streams, especially binary and low-cardinality ones, a pretrained LLM is the most accurate tier and is the right place to stop; do not assume the goal is to graduate to a task-trained model.
2. Check two cheap signals first. Label cardinality and how far a 100-example rule sits above chance predict which tier will win, before any training. High cardinality with a near-chance rule points to the task-trained tier; a binary task with a weak rule points to the LLM.
3. Graduate on a tie, not a win. The rule and a trained model are far cheaper to operate than an LLM (microseconds to milliseconds versus roughly half a second per call here), so moving to the cheaper tier as soon as it is statistically non-inferior captures most of the cost and latency benefit at no accuracy loss.
4. Measure per site rather than predict across sites. A learned predictor of the best tier from stream characteristics did not generalize at 21 datasets, so the reliable approach is to instrument each site and let the gate decide, not to guess from a model of the choice.

7 Limitations

The task-trained tier defaults to the cheap day-N head (TF-IDF, pixels, or MFCC features with logistic regression). To check whether the “LLM wins” results are an artifact of a weak baseline, we ran two stronger baselines. Frozen sentence-transformer embeddings (all-MiniLM-L6-v2) with logistic regression still lose to the LLM on short-text sentiment and moderation (sst2 0.80 vs 0.98; rotten_tomatoes 0.74 vs 0.94; tweet_hate 0.53 vs 0.77). A

fine-tuned DistilBERT (4k examples, 3 epochs) narrows the gap but does not close it (sst2 0.89 vs 0.98; imdb 0.87 vs 0.96, about 9 points). On high-cardinality intent, a tuned TF-IDF and logistic regression remained the strongest classical baseline (banking77 0.87, above fine-tuned DistilBERT’s 0.80 at this budget), consistent with ML winning there. The central finding therefore holds across baseline strength. Two caveats apply: the fine-tune was light and untuned, so heavier tuning could narrow but is unlikely to erase the sentiment gap, and frozen embeddings are not uniformly stronger (imdb MiniLM 0.78 is below TF-IDF’s 0.85 because of truncation). For image, a real baseline resolves the ambiguity: frozen ViT embeddings with logistic regression reach 0.95 on CIFAR-10, above the zero-shot vision LLM’s 0.78, so CIFAR’s apparent “LLM win” was the pixel-logreg strawman and image favors ML, as high-cardinality text does. The aligned table still lists the pixel-logreg tier; the ViT number is the robustness check.

Other limitations. Latency and cost are measured on a sample (latency_profile.json), and LLM latency is network-bound and will vary. We use one LLM (Claude Haiku) and one prompt, so absolute accuracies are sensitive to prompt and model; the claim is the tier ordering, not the exact numbers. As a robustness check we re-ran a second, weaker, free local LLM (qwen2.5:7b via Ollama) zero-shot on seven datasets: the tier ordering held on six (the trained model wins high-cardinality intent, the LLM wins binary sentiment), and the one flip (tweet_hate) is on-thesis, since the weaker LLM there drops below the trained model and cedes the harder task to it, shrinking the LLM’s winning region as model quality falls. The multimodal LLM tier is a proxy (spectrogram or sampled frames); native audio and video models are future work, and spectrogram-vision is shown to fail on environmental audio. The cross-site predictor does not generalize at this number of datasets (Section 4.4). With $n=400$ test subsets the accuracy confidence intervals are about ± 0.04 , and we use significance-aware tie bands accordingly.

8 Related work

Model cascades and LLM routing. Cascades route an input through progressively more expensive models and stop early when a cheap model is confident, a pattern dating to the Viola and Jones (2001) detector and recurring in modern LLM cost cascades such as FrugalGPT (Chen et al. 2023) and learned LLM routers (Ong et al. 2024). These methods reduce cost by choosing a model per query; they do not characterize, per classification site, whether a trained model should permanently replace the LLM, nor do they keep a deterministic rule as a safety floor.

Learning to defer. A predictor can abstain and defer to an expert when the expert is more likely correct (Madras et al. 2018; Mozannar and Sontag 2020). Deferral is a per-example decision against a fixed expert; AGA instead selects a per-stream terminal tier and graduates the whole site off the expensive tier once a cheaper one is statistically non-inferior.

Model selection and AutoML. Selecting an estimator and its hyperparameters from data is the subject of AutoML (Thornton et al. 2013; Hutter et al. 2019). That work selects within the trained-model family; our question is the prior one of which tier (rule, LLM, or trained model) should serve the stream at all, from cheap stream characteristics.

Production ML and statistical comparison. Brittle rules and silent regressions are documented sources of technical debt (Sculley et al. 2015; Breck et al. 2017). Champion-

challenger and shadow deployment compare a candidate against the incumbent before promotion, and the paired-McNemar test (McNemar 1947; Dietterich 1998) is the standard instrument for comparing two classifiers on the same data. AGA builds its gate on this test and adds the lifecycle-level guarantee of Section 4.1 and a cost-aware, multi-objective terminal selection.

Cost-sensitive and active learning. Cost-sensitive learning weights errors by cost (Elkan 2001) and active learning chooses which labels to acquire (Settles 2009). Both are complementary: AGA’s labeled-outcome axis is a cost the site already pays through its outcome stream, and the gate decides when enough has accumulated to graduate.

9 Broader Impact

This work helps practitioners decide, for each classification site, whether a rule, a large language model, or a trained model should serve it, and when to move between tiers. The potential positive consequences are lower inference cost and energy use, since routine classification can be graduated off always-on LLMs onto cheap local models; lower latency, which benefits on-device and real-time uses; and an auditable promotion process, since a deterministic rule is retained as a safety floor and a statistical gate is required before any change. The potential negative consequences, with mitigations, are as follows. An automated graduation policy could promote a worse decision-maker if the outcome labels feeding the gate are biased or sparse; the non-inferiority gate bounds but does not eliminate this risk, and the retained rule floor together with human review for safety-sensitive sites is the intended mitigation. The labels and outcomes used to train and gate the tiers can encode social biases, which this method does not address and which should be handled with standard fairness auditing. Finally, cutting cost is not a reason to remove human oversight from consequential decisions. This method only chooses which automated tier to use where automation is already warranted; it does not argue for automating decisions that should involve people.

10 Reproducibility

A single command, `scripts/reproduce_dmlr.sh`, regenerates every number. The MODEL stages skip automatically without an API key. See REPRODUCE.md.

11 References

- Breck, E., Cai, S., Nielsen, E., Salib, M., and Sculley, D. (2017). The ML Test Score: A Rubric for ML Production Readiness and Technical Debt Reduction. IEEE International Conference on Big Data.
- Chen, L., Zaharia, M., and Zou, J. (2023). FrugalGPT: How to Use Large Language Models While Reducing Cost and Improving Performance. Transactions on Machine Learning Research.
- Dietterich, T. G. (1998). Approximate Statistical Tests for Comparing Supervised Classification Learning Algorithms. Neural Computation, 10(7), 1895-1923.

- Elkan, C. (2001). The Foundations of Cost-Sensitive Learning. International Joint Conference on Artificial Intelligence.
- Foster, D. P. and Stine, R. A. (2008). Alpha-investing: a procedure for sequential control of expected false discoveries. *Journal of the Royal Statistical Society: Series B*, 70(2), 429-444.
- Hutter, F., Kotthoff, L., and Vanschoren, J., editors (2019). *Automated Machine Learning: Methods, Systems, Challenges*. Springer.
- Madras, D., Pitassi, T., and Zemel, R. (2018). Predict Responsibly: Improving Fairness and Accuracy by Learning to Defer. *Advances in Neural Information Processing Systems*.
- McNemar, Q. (1947). Note on the sampling error of the difference between correlated proportions or percentages. *Psychometrika*, 12(2), 153-157.
- Mozannar, H. and Sontag, D. (2020). Consistent Estimators for Learning to Defer to an Expert. *International Conference on Machine Learning*.
- Ong, I., Almahairi, A., Wu, V., Chiang, W.-L., Wu, T., Gonzalez, J. E., Kadous, M. W., and Stoica, I. (2024). RouteLLM: Learning to Route LLMs with Preference Data. [arXiv:2406.18665](https://arxiv.org/abs/2406.18665).
- Sculley, D., Holt, G., Golovin, D., Davydov, E., Phillips, T., Ebner, D., Chaudhary, V., Young, M., Crespo, J.-F., and Dennison, D. (2015). Hidden Technical Debt in Machine Learning Systems. *Advances in Neural Information Processing Systems*.
- Settles, B. (2009). *Active Learning Literature Survey*. University of Wisconsin-Madison, Computer Sciences Technical Report 1648.
- Thornton, C., Hutter, F., Hoos, H. H., and Leyton-Brown, K. (2013). Auto-WEKA: Combined Selection and Hyperparameter Optimization of Classification Algorithms. *ACM SIGKDD*.
- Viola, P. and Jones, M. (2001). Rapid Object Detection using a Boosted Cascade of Simple Features. *IEEE Conference on Computer Vision and Pattern Recognition*.