International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 04 , April - 2026

Predicting Digital Ad Performance the Hard Way: A Hybrid Stack Built from Public Kaggle Data and Hand-Tagged Indian Competitor Ads

Deepanshu Jindal, Medha Anand Chomal, Akashdeep Singla, Harsh Sinha, and Gurpreet Singh
Department of Computer Science / Chitkara University

Abstract - This paper addresses the challenge of pre-flight creative performance prediction for digital advertising campaigns. We assembled a hybrid dataset from nine Kaggle sources, three LLM-generated synthetic files, and 120 manually tagged competitor ads from the Meta Ad Library and Google Ads Transparency Centre (collected 20–28 April 2026). After pre-processing, the unified corpus comprises 21,643 campaign rows and 1,120 creative-metadata rows. We trained a stacked ensemble of HistGradientBoosting, LightGBM, XGBoost, and Random Forest with a Ridge meta-learner across 15 prediction tasks. Cross-validated R^2 reached 0.75 for CTR, 0.81 for CPC, and 0.75 for CPA. The high-CTR binary classifier achieved AUC = 0.994 with a Brier score of 0.033. Install-quality prediction attained $R^2 = 0.80$. An ablation study revealed two counter-intuitive findings: (a) stacking yields only marginal gains (~0.0006 R^2 on CTR) over a single HistGradientBoosting model, and (b) removing LLM-generated training rows improves D1 generalisation by 5.8 R^2 points. Both findings are reported in full.

Index Terms - click-through rate prediction, digital advertising, ensemble learning, gradient boosting, stacking meta-learner, LLM data augmentation, install quality prediction, ad performance modelling.

I. INTRODUCTION

A small scenario illustrates the motivation for this work. A three-person team managing a D2C baby-care brand holds INR 8 lakh in ad budget with Diwali ten days away. On a Sunday evening they lock in five creatives. By Wednesday, three are dead — each having spent INR 80,000 with CTR below the team's internal floor. The remaining two carry the entire campaign. Had a model been able to flag, on Sunday night, which two creatives were likely to perform, approximately INR 2.4 lakh in wasted spend could have been avoided.

The prediction problem is harder than it first appears. Outcome depends simultaneously on creative content, audience targeting, platform, and bid strategy. Most published CTR benchmarks (Criteo, Avazu) hash creative content out of the feature set, eliminating the very lever that practitioners can control. Furthermore, marketing teams require model explanations that map to actionable decisions; deep neural networks that deliver marginal accuracy gains at the cost of interpretability offer limited practical value [9].

We held ourselves to three constraints: (1) *Reproducibility* — all data sources are public or documented; (2) *Actionability* — prediction targets are metrics a media buyer already uses; (3) *Honesty* — every R^2 is reported with a 95% bootstrap CI and a five-fold CV score, and negative ablation findings are reported, not suppressed.

II. RELATED WORK

CTR prediction has a 20-year lineage, beginning with logistic regression on sparse hashed feature vectors at early ad networks. The field progressed through factorisation machines [FM], field-aware factorisation machines [FFM], and latterly deep cross-networks deployed at scale by major platforms. Criteo and Avazu remain the canonical open benchmarks. Their value lies in scale; their limitation is that creative content is hashed away, removing the marketing-side use case we address.

Post-install retention is less well studied in the open literature. Mobile measurement partners (AppsFlyer, Adjust) instrument SDKs that emit D1, D7 and D30 retention windows. Those tables sit behind vendor agreements and no public Kaggle source emits real post-install retention at the ad level. We document our heuristic workaround in Section III and quantify its cost in the ablation (Section V-A).

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 04 , April - 2026

III. DATA AND PRE-PROCESSING

A. Sources

Table I summarises the ten data sources. Sources 1–6 are public Kaggle datasets providing real campaign-level metrics. Sources 7–9 are creative-metadata files synthesised by ChatGPT, Claude, and Gemini respectively. Source 10 is a manual tag file covering 120 competitor ads from 20 Indian brands across baby-care, parenting, and edtech verticals (Mamaearth, FirstCry, Pampers, BabyChakra, Healofy, Huggies, Himalaya, Johnson's, Sebamed, MamyPoko, Meesho, Flipkart, Amazon India, BYJU's Early Learn, Khan Academy Kids, Pediasure, Cetaphil Baby, The Moms Co, Mother Sparsh, and Chicco), collected between 20 and 28 April 2026.

#	Source	Rows	Type
1	Kaggle Facebook Ad Campaigns [1]	1,143	Real
2	Kaggle Social Media Advertising 300k [2]	10,000 (sampled)	Real
3	Kaggle Ad Click Prediction 10k [3]	10,000	Real
4	Kaggle Social Media Optimisation [4]	500	Real
5	Kaggle Ad Campaign Relational DB [5]	400,000 events → aggregated	Real
6	Kaggle Marketing Campaign 200k [6]	10,000 (sampled)	Real
7	ChatGPT-generated creative metadata	500	LLM
8	Claude-generated edge cases	200	LLM
9	Gemini-generated competitor metadata	300	LLM
10	Manual: Meta Ad Library [7] + Google Ads Transparency [8]	120	Observed

TABLE I. Data sources used in this study.

B. Pre-processing Issues

Several non-trivial cleaning challenges were encountered:

(1) Source 1 contained 382 rows with columns shifted one position due to an unquoted comma in the campaign name field. Detected via unexpected string values in the campaign_id column; diagnosis required approximately half a day.

(2) Sources 2 and 6 store monetary spend as '$xx.xx' strings. Failure to strip the dollar sign before casting causes Pandas to silently coerce the entire column to NaN.

(3) Source 5 contains no per-ad metrics. The 400,000 event rows were aggregated by ad_id to recover impression, click, and conversion totals.

(4) The three LLM-generated files (Sources 7–9) used inconsistent column naming conventions. A small mapping layer was written to harmonise them.

(5) The manual tag file used eight creative theme categories; the LLM files used thirteen. A unified taxonomy of eight categories was adopted by collapsing the additional LLM categories, at the cost of some granularity.

C. Final Tables

After cleaning, two tables were produced: Table A (real campaign data), containing 21,643 rows and 12 columns; and Table B (creative metadata), containing 1,120 rows from Sources 7–10. Categorical features are label-encoded; missing numeric values are filled with −1 to allow tree-based learners to split on absence rather than imputing a mean.

D. Leakage Audit

A critical leakage problem was discovered during initial modelling. The first CPM model returned $R^2 = 1.0000$. Investigation revealed that log_spend and log_impressions were included as features while CPM = (spend × 1000) / impressions was the target; any tree learner can reconstruct the target via arithmetic. Removing log_spend reduced R^2 only marginally, as platform encoding alone was sufficient to determine CPM in our dataset (Pinterest CPM ≈ $192 vs. Meta/Twitter/Instagram ≈ $124). Removing platform encoding from CPM features produced a final R^2 of 0.73 — a genuine prediction. Per-target feature exclusions are documented in the released training script.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 04 , April - 2026

IV. METHOD

The pipeline employs a stacked ensemble of four base learners — HistGradientBoosting (HGB) [12], LightGBM [10], XGBoost [11], and Random Forest — with a Ridge meta-learner ($\alpha = 0.5$). Outliers are clipped at the 0.5th and 99.5th percentiles per target. Model evaluation uses a held-out 20% test split, five-fold cross-validation, and 200-resample bootstrap confidence intervals on R^2.

A. Algorithm

Algorithm 1: Stacked-Ensemble Training
INPUT: X (n×d feature matrix), y (target vector), B (base learners)
OUTPUT: M (final stacked predictor)

1. Partition rows into 5 disjoint folds $F_1, ..., F_5$
2. FOR each base learner $b \in B$:
   FOR each fold $k \in \{1..5\}$:
   Train b on rows NOT in F_k
   Record b's predictions on F_k as the k-th block of Z[:, b]
3. Train each $b \in B$ on the full training set X
4. Train Ridge meta-learner R on (Z, y) with $\alpha = 0.5$
5. At inference: build $z = [b_1(x), b_2(x), b_3(x), b_4(x)]$ and return R(z)

B. Mathematical Formulation

The stacked predictor combines base learner outputs via a Ridge meta-learner:

$$\hat{y}(x) = \Sigma_{b\in\{hgb,lgb,xgb,rf\}}\, \alpha b \cdot b(x) + \alpha_0 \quad \text{..... (1)}$$

The Ridge meta-learner minimises:

$$\min_{\alpha} \|y - Z\cdot\alpha\|^2 + \lambda\|\alpha\|^2, \quad \lambda = 0.5 \quad \text{..... (2)}$$

Performance is measured by the coefficient of determination on the held-out test set:

$$R^2 = 1 - \Sigma_i(y_i - \hat{y}_i)^2 / \Sigma_i(y_i - \bar{y})^2 \quad \text{..... (3)}$$

For binary classification tasks, the Brier score [13] and AUC are reported:

$$Brier = (1/n)\, \Sigma_i\, (\hat{p}_i - y_i)^2 \quad \text{..... (4)}$$

$$AUC = \int_0^1 TPR(t)\, d(FPR(t)) \quad \text{..... (5)}$$

Permutation feature importance is averaged over five repeats:

$$PI_j = R^2(model, X) - mean_{r=1..5}\, R^2(model, X \text{ with column } j \text{ shuffled}) \quad \text{..... (6)}$$

Bootstrap 95% confidence intervals on R^2 are computed from 200 resamples:

$$CI_{95}(R^2) = [P_{2.5}, P_{97.5}] \text{ of } \{R^2(y^*b, \hat{y}^*b) : b = 1..200\} \quad \text{..... (7)}$$

Mean absolute percentage error serves as a sanity check on multiplicative targets:

$$MAPE = (1/n)\, \Sigma_i\, |y_i - \hat{y}_i| / |y_i| \quad \text{..... (8)}$$

V. RESULTS

Table II reports all 15 prediction tasks. R^2 values, 95% bootstrap confidence intervals, mean absolute error, and five-fold CV statistics are given for regression tasks. Accuracy, F1, AUC, and Brier score are given for classification tasks.

Task	Type	Headline	Detail / CI95	Error	CV5 mean±std	N(test)
CTR	Reg	R^2 0.733	[0.70, 0.76]	MAE 0.034	0.747±0.014	4,329
CPC	Reg	R^2 0.806	[0.80, 0.82]	MAE 2.075	0.809±0.003	3,492
CVR	Reg	R^2 0.678	[0.60, 0.74]	MAE 0.039	0.650±0.038	4,275
ROI	Reg	R^2 0.332	[0.31, 0.36]	MAE 1.595	0.336±0.014	2,166

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 04 , April - 2026

Task	Type	Headline	Detail / CI95	Error	CV5 mean±std	N(test)
CPM	Reg	R^2 0.733	[0.71, 0.75]	MAE 20.338	0.728±0.006	2,229
CPA	Reg	R^2 0.767	[0.75, 0.78]	MAE 33.978	0.752±0.011	3,523
Engagement Score	Reg	R^2 0.387	[0.35, 0.41]	MAE 1.854	0.392±0.019	2,100
High CTR Clf.	Clf	Acc 0.950	F1 0.907 | AUC 0.994	Brier 0.033	0.913±0.003	4,329
High Engagement Clf.	Clf	Acc 0.452	F1 0.464 | AUC 0.635	Brier 0.224	0.052±0.010	4,329
High CVR Clf.	Clf	Acc 0.635	F1 0.307 | AUC 0.550	Brier 0.235	0.157±0.018	4,275
D1 Retention Rate	Reg	R^2 0.729	[0.52, 0.84]	MAE 2.075	0.723±0.030	224
D7 Retention Rate	Reg	R^2 0.647	[0.54, 0.72]	MAE 0.018	0.660±0.026	224
D30 Retention Rate	Reg	R^2 0.685	[0.60, 0.74]	MAE 0.012	0.695±0.023	224
Install Quality Score	Reg	R^2 0.799	[0.72, 0.85]	MAE 5.648	0.738±0.031	224
Cross-Platform Lift	Reg	R^2 0.368	[0.19, 0.51]	MAE 0.068	0.383±0.054	224

TABLE II. Results for all 15 prediction tasks.

The two most informative tasks in terms of practitioner utility are CTR (CV $R^2 = 0.747 \pm 0.014$) and CPC (CV $R^2 = 0.809 \pm 0.003$). The high-CTR binary classifier is both discriminative (AUC = 0.994) and calibrated (Brier = 0.033), meaning the predicted probability can be used directly as a confidence score.

Permutation importance reveals that CTR is dominated by source identity, log-impressions, and platform identity — consistent with the leakage audit findings. D1 retention, in contrast, is driven by source dataset, a has_offer flag, creative theme, and audience segment. This confirms a well-known practitioner heurism: discount-led creatives purchase clicks; expert-led creatives purchase retained users.

A. Ablation Study

Table III presents the CTR ablation and Table IV presents the D1 retention ablation.

Configuration	R^2 (CTR)	Δ vs Full
Full 4-model stack	0.7325	—
Drop HGB	0.7290	−0.0036
Drop LightGBM	0.7332	+0.0007
Drop XGBoost	0.7316	−0.0009
Drop Random Forest	0.7327	+0.0001
Only HGB	0.7320	—
Only LightGBM	0.7252	—
Only XGBoost	0.7275	—
Only Random Forest	0.7259	—

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 04 , April - 2026

TABLE III. Model ablation on CTR prediction.

Configuration	R^2 (D1)	Δ vs Full
Full 4-model stack	0.7199	—
Drop HGB	0.7355	+0.0157
Drop LightGBM	0.7467	+0.0269
Drop XGBoost	0.7654	+0.0455
Drop Random Forest	0.7009	−0.0190
Drop manual tags (D1)	0.7541	+0.0343
Drop LLM rows (D1)	0.7778	+0.0579

TABLE IV. Model and data ablation on D1 retention.

Two findings emerge. First, on CTR, all four base learners achieve within 0.01 R^2 of one another; the full stack improves on the best single learner (HGB, $R^2 = 0.7320$) by only 0.0006 — within measurement noise. Second, removing the LLM-augmented rows from the D1 training set *improves* R^2 by 0.0579, reproducible across three independent random seeds. The most plausible explanation is that the heuristic used to synthesise missing D1 labels in the LLM files introduced a deterministic artefact that the model fitted, and this pattern did not transfer to the manually-tagged ground truth.

B. Baseline Comparison

Model	R^2 (CTR)	R^2 (D1)
Predict Mean	−0.0015	−0.0104
Linear Regression	0.2105	0.1325
Ridge Regression	0.2105	0.1340
Single HGB	0.7319	0.7429
Single XGBoost	0.7271	0.6936
Full Stack (4 learners)	0.7325	0.7199

TABLE V. Baseline comparison on CTR and D1 retention.

Predicting the training mean yields R^2 near zero, confirming the variance in both targets is real. Linear and Ridge regression capture only 21% of CTR variance and 13% of D1 variance, confirming that non-linear feature interactions are essential. Single HGB and single XGBoost perform comparably to the full four-model stack on CTR. Practitioner recommendation: begin with a single HGB or XGBoost; add stacking complexity only if the single model plateaus on additional tasks.

VI. LIMITATIONS

Several limitations constrain the scope of conclusions that can be drawn from this work.

• *Source heterogeneity.* Six Kaggle vendors define impression and click differently. Source identity is included as a feature to allow learners to absorb systematic bias; residual confounding is likely not fully eliminated.

• *Synthetic retention labels.* D7, D30, and cross-platform lift are heuristically derived, not sourced from a measurement partner. The ablation quantifies one cost: 5.8 R^2 points lost on D1 from LLM-generated rows.

• *Manual sample size.* 120 ads provide categorical breadth but are insufficient for brand-level inference. A target of ~500 ads per priority brand, stratified across platforms, is recommended for a follow-up collection.

• *ROI ceiling.* ROI in Source 6 has $|r| < 0.02$ with every numeric feature in the dataset; prediction is not feasible without better underlying data.

• *Stack complexity.* As demonstrated in Section V-B, stacking is not strictly necessary for campaign-level targets. It was retained because the ensemble methodology was a demonstration goal of the project and provides 1–2 R^2 points of benefit on noisier creative-metadata targets.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 04 , April - 2026

VII. CONCLUSION

This paper presented a hybrid data pipeline and stacked ensemble for predicting 15 digital advertising performance metrics. Cross-validated R^2 ranged from 0.65 to 0.81 on well-defined campaign targets. The high-CTR classifier achieved AUC = 0.994 and a Brier score of 0.033. D1 and install-quality predictions attained R^2 between 0.73 and 0.80.

Two findings are highlighted for practitioners. First, a single HistGradientBoosting model provides performance within noise of the full four-model stack on campaign-level targets, offering a simpler and more maintainable deployment option. Second, LLM-generated synthetic rows can degrade generalisation when the synthesiser introduces label artefacts; augmentation quality must be validated before inclusion in training sets.

Two avenues for future work are identified. Replacing the heuristic retention block with real measurement-partner data would remove the largest source of label noise in the creative-metadata targets. Replacing categorical theme labels with LLM embeddings of actual ad copy text would capture richer creative signal than the current taxonomy-based approach.

REFERENCES

[1] Kaggle. "Facebook Ad Campaign Dataset." kaggle.com/datasets, accessed Apr. 2026.
[2] Kaggle. "Social Media Advertising 300k." kaggle.com/datasets, accessed Apr. 2026.
[3] Kaggle. "Ad Click Prediction 10k." kaggle.com/datasets, accessed Apr. 2026.
[4] Kaggle. "Social Media Ad Optimisation." kaggle.com/datasets, accessed Apr. 2026.
[5] Kaggle. "Full Ad Campaign Relational Database." kaggle.com/datasets, accessed Apr. 2026.
[6] Kaggle. "Marketing Campaign Performance 200k." kaggle.com/datasets, accessed Apr. 2026.
[7] Meta Platforms, Inc. "Meta Ad Library." facebook.com/ads/library, accessed Apr. 2026.
[8] Google LLC. "Google Ads Transparency Centre." adstransparency.google.com, accessed Apr. 2026.
[9] J. H. Friedman, "Greedy function approximation: a gradient boosting machine," Annals of Statistics, vol. 29, no. 5, pp. 1189–1232, 2001.
[10] G. Ke, Q. Meng, T. Finley et al., "LightGBM: A highly efficient gradient boosting decision tree," in Proc. NeurIPS, vol. 30, 2017.
[11] T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," in Proc. KDD, pp. 785–794, 2016.
[12] F. Pedregosa et al., "Scikit-learn: Machine learning in Python," Journal of Machine Learning Research, vol. 12, pp. 2825–2830, 2011.
[13] G. W. Brier, "Verification of forecasts expressed in terms of probability," Monthly Weather Review, vol. 78, no. 1, pp. 1–3, 1950.