Available online at website: https://jurnal.iaii.or.id/index.php/RESTI

JURNAL RESTI
(Rekayasa Sistem dan Teknologi Informasi)
Vol. 10 No. 5 (2026) 1243 - 1251 e-ISSN: 2580-0760

Heterogeneous Stacking Ensemble with SMOTE-Based Oversampling for Agricultural Commodity Price Prediction under Extreme Class Imbalance

Yessica Siagian[1], Neni Mulyani[2], Jeperson Hutahaean[3], Dinda Maharani[4], Jenny Erlita[5]
[1,2,3,4,5]Faculty of Computer Science, Information System Program, Universitas Royal, Kisaran, Indonesia

*yessica.cyg123@gmail.com

Abstract

Predicting agricultural commodity prices is vital for food security and market stability, but price data is highly non-linear, high-cardinality, and severely imbalanced, rendering conventional accuracy metrics misleading. This study proposes a heterogeneous stacking ensemble of LightGBM, XGBoost, and Random Forest base learners with a LightGBM meta-learner, evaluated against five BorderlineSMOTE oversampling variants and four ensemble configurations under a 282:1 class imbalance. A controlled ablation across six configurations — including a new factorial design that isolates the contributions of oversampling and an MLP base learner — demonstrates that performance is driven by the stacking architecture rather than by synthetic oversampling or deep learning components. The ensemble without oversampling achieves MCC = 0.9509 and AUC-OvR = 0.9908 on the held-out test set, while the addition of an MLP base learner yields no consistent improvement (|ΔMCC| < 0.008 across both SMOTE and no-SMOTE conditions). These findings identify cost-sensitive heterogeneous stacking as an effective and computationally tractable strategy for highly imbalanced tabular agricultural data, and establish that synthetic oversampling and MLP-based components are unnecessary under these conditions.

Keywords: agricultural commodity price prediction; deep learning; LightGBM; SMOTE; stacking ensemble

How to Cite: Y. Siagian, N. . Mulyani, J. . Hutahaean, D. . Maharani, and J. . Erlita, "Heterogeneous Stacking Ensemble with SMOTE-Based Oversampling for Agricultural Commodity Price Prediction under Extreme Class Imbalance", *J. RESTI (Rekayasa Sist. Teknol. Inf.)* , vol. 10, no. 5, pp. 1243–1251, Oct. 2026, doi: 10.29207/resti.v10i5.8065.
*Permalink/*DOI: https://doi.org/10.29207/resti.v10i5.8065

Received: June 12, 2026
Accepted: September 5, 2026
Available Online: October 1, 2026

This is an open-access article under the CC BY 4.0 License
Published by Ikatan Ahli Informatika Indonesia

1. Introduction

Agricultural commodity price prediction is crucial for food security and economic stability in agrarian societies [1-3]. Price fluctuations in staple crops create significant financial uncertainty for farmers, traders, and policymakers [4], [5]. Accurate, timely price forecasts help producers optimize harvest and marketing, enable intermediaries to manage inventory risk, and guide government measures to stabilize volatile markets.

In developing economies, where agriculture is a major share of GDP and rural livelihoods depend on commodity revenues, unexpected price swings have severe social and economic impacts [6]. Robust models that capture the non-linear dynamics of agricultural prices are therefore critical for sustainable food systems and evidence-based market policy. Despite their importance, agricultural price data pose major statistical challenges that undermine conventional modeling. The daily commodity price dataset in this study contains 5,552 records across 23 Indian states, 245 districts, 531 markets, 186 commodities, and 293 variety labels, resulting in high-cardinality, sparse feature spaces. Research on agricultural price prediction has progressed through several methodological generations [7-9]. Time-series models [10], [11], notably ARIMA and its seasonal variants [12-14], have long been used for commodity forecasting due to their interpretability. However, their linearity assumptions limit their ability to capture the non-stationary, volatile dynamics of agricultural markets.

Support vector machines enabled non-linear prediction but perform poorly in high-dimensional, imbalanced feature spaces [15], [16]. Ensemble methods like Random Forests and gradient-boosted trees achieve strong accuracy on tabular data yet still suffer from majority-class dominance under extreme imbalance [17].

Deep learning models like multilayer perceptrons and recurrent networks learn rich features but require large,

balanced datasets and remain hard for practitioners to interpret [18], [19]. Synthetic oversampling methods such as SMOTE and its adaptive variants address class imbalance in general classification tasks [20].

However, no prior study in agricultural commodity price prediction has jointly applied adaptive SMOTE variant selection, heterogeneous stacking of machine learning and deep learning base learners, and a systematic ablation study isolating each component's contribution.

This research proposes Stack-SMOTE-AgriPrice: a seven-step preprocessing pipeline, BorderlineSMOTE-based class balancing, and a heterogeneous stacking ensemble. The ensemble uses four base learners—LightGBM, XGBoost, Random Forest, and a multilayer perceptron—trained with stratified five-fold cross-validation to generate out-of-fold (OOF) probability estimates, which are then combined by a LightGBM meta-learner. This research introduces a seven-step preprocessing workflow for agricultural commodity price data that uses log1p transformation, winsorization at the 1st and 99th percentiles, and target encoding to address right-skewed distributions and high-cardinality categorical variables. It systematically benchmarks five SMOTE variants, SMOTE, BorderlineSMOTE, ADASYN, SMOTETomek, and SMOTEENN, to find the most effective oversampling method for extreme class imbalance. This research is novel in combining LightGBM, XGBoost, Random Forest, and MLP as base learners with LightGBM as a meta-learner, unifying machine learning and deep learning models in a single predictive layer. A systematic ablation study was performed to assess each component's contribution and to evaluate Grade classification performance using standard metrics.

2. Methods

The dataset used in this study is a daily commodity price dataset with transaction records from agricultural wholesale markets across India.

Table 1. Dataset Characteristics

Property	Value
Total records (raw)	5,557
Total records (after cleaning)	5,552
Records removed (quality violations)	5
Categorical features	5: State, District, Market, Commodity, Variety
Numerical features	3: Min Price, Max Price, Modal Price
Classification target (Grade)	4 classes: Small (0.31%), Medium (12.18%), Large (1.10%), FAQ (86.40%)
Regression target	Modal Price (INR per quintal)
Class imbalance ratio (FAQ: Small)	282: 1
Geographic coverage	23 states, 245 districts, 531 markets
Training set	4,441 records (80%)
Test set	1,111 records (20%)

The raw data contained 5,557 records. After quality checks, five records were removed for domain-validity violations, zero or negative prices, Modal Price below one INR, or Modal Price higher than the reported Max Price, resulting in 5,552 observations.

Table 1 summarizes the dataset's main characteristics. Figure 1 shows the seven-step preprocessing pipeline applied before model training. Steps were executed in a fixed order, with all scalers fit only on the training partition to prevent test-set data leakage.

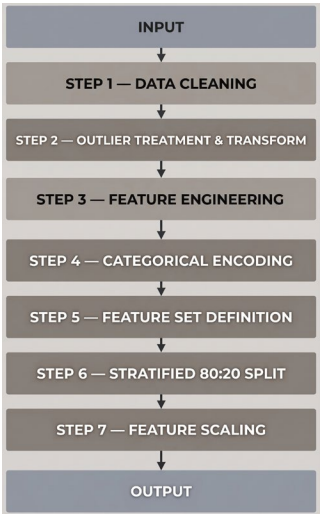


Figure 1. Data Preprocessing Pipeline

2.1 SMOTE-Based Class Imbalance Handling

SMOTE creates synthetic minority samples via linear interpolation in feature space [21]. For each minority instance, it finds k same-class nearest neighbors and generates a new sample by interpolating between the instance and a randomly chosen neighbor using a uniform random factor in [0, 1]. This increases minority representation without duplicating observations and helps reduce decision-boundary distortion caused by majority-class dominance.

Borderline SMOTE refines this approach by generating synthetic samples only for minority instances near the decision boundary [21-23]. Focusing on these boundary regions yields more informative synthetic examples where classification is hardest, rather than in interior minority regions that are already well classified. Given the 282:1 class imbalance, interior minority samples are too sparse for effective synthesis, so focusing on boundary-region generation is more efficient.

SMOTE is applied only to the training set, never to the full dataset before splitting or to the test set. With cross-validation, SMOTE is rerun separately on each fold's training subset to keep validation splits unbiased. Five SMOTE variants were benchmarked: SMOTE [21], BorderlineSMOTE [24], ADASYN [25], SMOTETomek [26], and SMOTEENN [27].

2.2 Stacking Ensemble Architecture

Stacked generalization builds a meta-learning layer that learns to optimally combine multiple base learners [28],

[29]. It uses out-of-fold (OOF) predictions: each base learner is trained on some folds and evaluated on the held-out fold, so the meta-learner is trained only on predictions from data unseen by the base learner [30].

For a dataset D split into T stratified folds {D_1, …, D_T}, the OOF probability vector for a sample x_i in fold D_t is computed as in equation. The collection of these OOF probability vectors from all K base learners forms the meta-feature set used to train the meta-learner: $\hat{z}_i = f_t(x_i)$, where f_t is trained on $D \setminus D_t$.

LightGBM uses leaf-wise tree growth with gradient-based one-sided sampling for efficient training on large tabular data [31], [32]. Class imbalance is addressed via `class_weight='balanced'`, which sets sample weights inversely to class frequency. The model uses 500 estimators, a 0.05 learning rate, max depth 6, 63 leaves per tree, and 0.8 subsample and column subsample ratios.

XGBoost grows trees level-wise with L1 and L2 regularization, complementing LightGBM's structural bias and resisting overfitting on sparse high-cardinality features [33], [34]. The base learner uses 400 estimators, a 0.05 learning rate, maximum depth 6, and 0.8 subsampling.

Random Forest builds an ensemble of decorrelated decision trees via bootstrap aggregation and random feature subsets [35], yielding reduced variance and interpretable feature importance. Class imbalance is addressed with class_weight='balanced'. The model uses 200 trees, maximum depth 12, and a minimum of 2 samples per leaf.

The multilayer perceptron (MLP) base learner provides qualitatively different probability surfaces via deep non-linear feature transformations [36], [37], adding representational diversity beyond tree-based models. It uses three fully connected layers (128 → 64 → 32 neurons) with ReLU activations, trained with Adam (learning_rate_init = 0.001, batch_size = 256) and early stopping (patience = 20, max_iter = 300). Unlike the tree-based learners, the MLP uses MinMax-scaled inputs to avoid activation saturation.

All OOF predictions used Stratified 5-Fold cross-validation (n_splits=5, shuffle=True, random_state=42), preserving the training set's class distribution in each fold. Using stack_method='predict_proba' passed four-dimensional probability vectors instead of hard labels as meta-features, retaining full probabilistic information for the meta-learner.

The architecture is used for two prediction tasks: a StackingClassifier for Grade classification across four imbalanced quality classes, and a StackingRegressor for Modal Price prediction, with Ridge regression (alpha = 1.0) as the meta-learner. Figure 2 shows the full stacking ensemble.

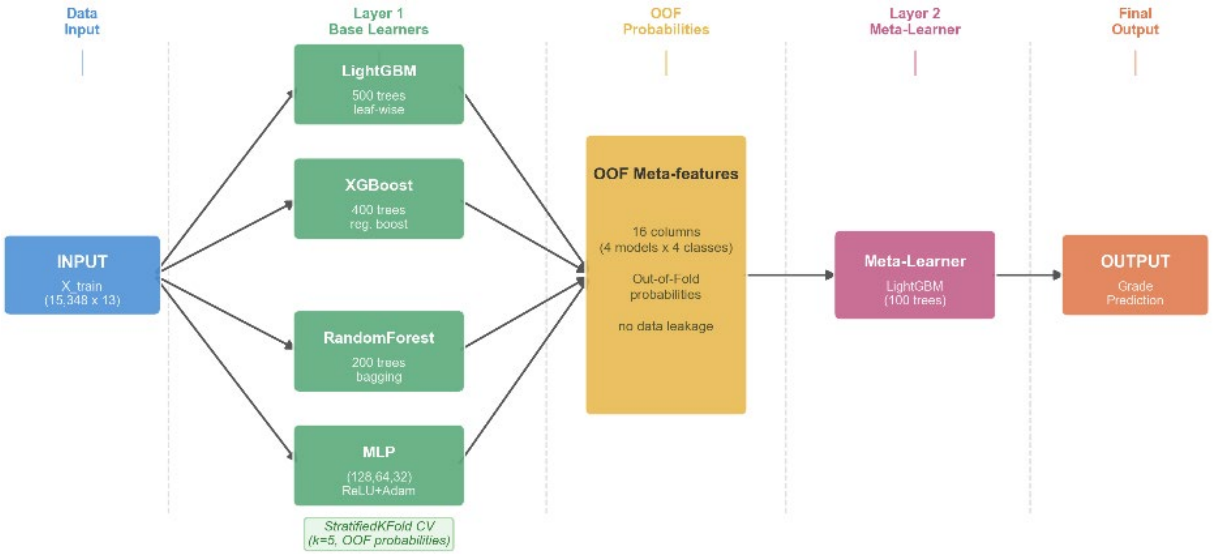


Figure 2. Proposed Heterogeneous Stacking Ensemble Architecture

2.3. Evaluation Framework

All experiments used Python 3.12 with scikit-learn 1.5 for classical and ensemble models, LightGBM 4.x, XGBoost 2.x, and imbalanced-learn 0.12 for SMOTE variants. Training times are reported per model configuration for reproducibility.

Model generalization was evaluated with Stratified 5-Fold cross-validation on the training set (StratifiedKFold, n_splits = 5, shuffle = True, random_state = 42). Stratification is crucial given the 282:1 class imbalance; without it, some folds could lack the rarest class (Small, n_train = 14), producing unreliable fold-level estimates.

For grade classification, we use macro-averaged F1 (F1-Macro) as the main metric. F1-Macro computes the precision–recall harmonic mean for each class and then averages them equally, so errors on Small and Large are penalized as much as errors on FAQ.

Secondary metrics are the geometric mean of per-class recall (G-Mean), the Matthews Correlation Coefficient

(MCC) which uses all confusion-matrix cells and is robust to severe imbalance and the macro-averaged one-vs-rest ROC AUC (AUC-OvR) [38], [39]. For Modal Price regression, all metrics are computed in the original INR space by applying the inverse log1p (expm1) to model outputs and ground truth. Reported metrics are RMSE, MAE, R^2, and MAPE.

Eleven baseline classifiers are grouped into two tiers. Tier 1 includes five classical models: Decision Tree, Gaussian Naive Bayes, K-Nearest Neighbors (k = 5), Logistic Regression (max_iter = 500), and an RBF-kernel SVM. Tier 2 includes five strong models: Random Forest, XGBoost, LightGBM, MLP, and Extra Trees. All baselines were trained with BorderlineSMOTE applied to the training set.

3. Results and Discussions

BorderlineSMOTE achieved the best F1-Macro (0.6263) and AUC-OvR (0.9467). ADASYN obtained the highest G-Mean (0.7727) and the shortest training time (0.325 s), while standard SMOTE and SMOTETomek reached F1-Macro scores of 0.5743 and 0.5866, respectively.

SMOTEENN yielded the lowest F1-Macro (0.4941) despite the longest training time (0.731 s), due to its aggressive Edited Nearest Neighbor cleaning, which removed many majority-class boundary samples and reduced the dataset to 12,607 observations, far fewer than the 15,348 samples from SMOTE and BorderlineSMOTE. Table 2 compares five SMOTE variants using a LightGBM classifier (class_weight = 'balanced') on the held-out test set (n = 1,111).

The performance boost of BorderlineSMOTE over standard SMOTE (ΔF1-Macro = +0.0520, ΔAUC-OvR = +0.0166) directly reflects its smarter design by generating synthetic samples only for minority instances near the decision boundary. It zeroes in on the most hotly contested regions of the feature space where better discrimination matters most. At the extreme imbalance ratio of 282:1, minority instances occupy a narrow feature-space region. Standard SMOTE spreads synthetic samples uniformly there, whereas BorderlineSMOTE focuses on the decision boundary, where new samples most improve classifier learning.

Table 2. Comparison of Five Smote Oversampling Techniques

Technique	F1-Macro	G-Mean	MCC	AUC-OvR	Time (s)
SMOTE	0.5743	0.7685	0.8419	0.9301	0.348
BorderlineSMOTE	**0.6263**	0.7583	**0.8663**	**0.9467**	0.349
ADASYN	0.5777	**0.7727**	0.8565	0.9233	**0.325**
SMOTETomek	0.5866	0.7682	0.8442	0.9267	0.669
SMOTEENN	0.4941	0.7499	0.6878	0.8806	0.731

Table 2 shows the computational trade-off is modest: BorderlineSMOTE (0.349 s) and SMOTE (0.348 s) have nearly identical training times. ADASYN yields a similar F1-Macro (0.5777) in less time (0.325 s) but lags BorderlineSMOTE by 0.0486 in F1-Macro and 0.0234 in AUC-OvR, supporting the small extra cost. SMOTETomek (0.669 s) and SMOTEENN (0.731 s) take much longer without proportional performance gains. Thus, BorderlineSMOTE was chosen for all subsequent experiments.

3.2 Baseline and Individual Model Performance

Among the five Tier 1 classical models, Decision Tree achieved the highest F1-Macro (0.4000), followed by K-Nearest Neighbors (0.3004), Gaussian Naive Bayes (0.2293), Logistic Regression (0.2288), and SVC (0.2122).

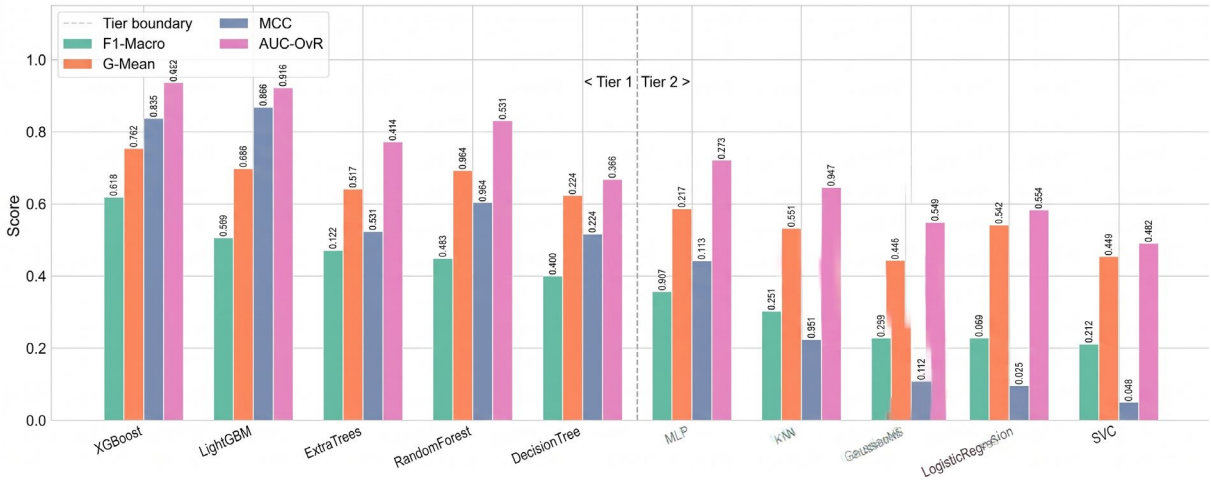


Figure 3. Proposed Heterogeneous Stacking Ensemble Architecture

Decision Tree's stronger performance indicates that commodity Grade classes are non-linearly separable, making linear (Logistic Regression) and feature-independence (Naive Bayes) assumptions poorly suited

to this high-cardinality, non-linear feature space. SVC achieved the lowest Tier 1 F1-Macro (0.2122) despite a 47.835 s training time, indicating that RBF kernel optimization fails to find an effective boundary under extreme imbalance without cost-sensitive adjustments.

Tier 2 ensemble and deep learning models clearly outperformed Tier 1. XGBoost achieved the best F1-Macro (0.6165) and G-Mean (0.7526) among all baselines in 1.326 s, highlighting the effectiveness of gradient boosting with explicit regularization for tabular imbalanced classification.

LightGBM achieved the second-highest F1-Macro (0.5001) and the highest MCC (0.8687), indicating strong discrimination. Extra Trees (0.4717) and Random Forest (0.4503) performed moderately, while MLP had the lowest F1-Macro among Tier 2 models (0.3575), highlighting the challenge of training neural networks on highly imbalanced tabular data with few minority samples.

Figure 3 compares F1-Macro, MCC, and AUC-OvR across all baseline models. Among individual models, XGBoost attained the highest F1-Macro (0.6165), making it the best F1-based baseline, while LightGBM achieved the highest MCC (0.8687). This shows that the choice of primary metric strongly affects model ranking under extreme class imbalance.

3.3 Proposed Stacking Ensemble Results

The stacking ensemble achieved an F1-Macro of 0.4778, an MCC of 0.8819, and an AUC-OvR of 0.9534 on the test set. It outperformed the best individual baselines, improving MCC over LightGBM (0.8687) by 0.0132 and AUC-OvR over XGBoost (0.9367) by 0.0167.

These results show that the stacking ensemble offers better probabilistic discrimination and calibration than any single model. However, XGBoost achieved a much higher F1-Macro (0.6165 vs. 0.4778, Δ = -0.1387), indicating that the proposed method mainly improves imbalance-robust metrics rather than hard-label classification performance.

Per-class analysis shows uneven classification quality. Medium (n_test = 136) and FAQ (n_test = 960) achieved F1-scores of 0.9214 and 0.9896, indicating near-perfect performance for the largest classes. In contrast, Small (n_test = 3) and Large (n_test = 12) had F1-scores of 0.000, due to very limited test support rather than systematic model failure.

Figure 4 presents the confusion matrix of the proposed stacking ensemble on the full test set (n = 1,111). It reveals strong performance for the dominant classes, with high correct-classification rates for FAQ and Medium. In contrast, the complete lack of correctly classified small and large instances underscores the instability and unreliability of F1 estimates when class support is extremely limited.

Figure 5 showcases the ROC curves for each Grade class under the One-vs-Rest (OvR) evaluation scheme, revealing robust classification performance. The macro-averaged AUC-OvR of 0.9534 highlights excellent probabilistic discrimination across all four classes.

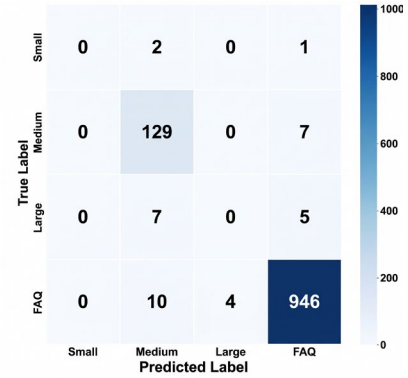


Figure 4. Confusion Matrix of the Proposed Stacking Ensemble on Test Set

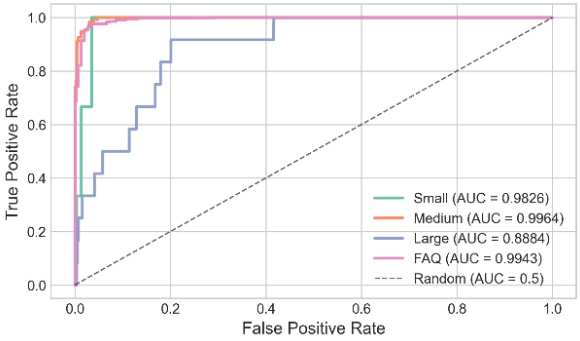


Figure 5. ROC Curves (One-vs-Rest) of the Proposed Stacking Ensemble

3.4 Ablation Study

Table 3 reports results across six configurations designed as a 2×2 factorial experiment over two binary factors supplemented by two single-component baselines (Configurations C and D). All configurations share the same preprocessing pipeline, training–test partition, and random seed to ensure that observed differences are attributable solely to the architectural factor under examination.

Configurations E and F, added in this revision, complete the factorial design by isolating the MLP base learner's contribution under both data conditions. Reviewer B correctly noted that the MLP's standalone F1-Macro of 0.3575 was the weakest among all base learners, and that no ablation existed to quantify its marginal ensemble contribution. Under the SMOTE condition, removing MLP from the stacking layer (Configuration A vs. Configuration E) reduces MCC by 0.0070 and AUC-OvR by 0.0047. Under the no-SMOTE condition, removing MLP (Configuration B vs. Configuration F) increases MCC by 0.0075 while leaving AUC-OvR virtually unchanged (Δ = +0.0002). The effect is below 0.008 in MCC and 0.005 in AUC-

OvR in both conditions, and its direction reverses depending on whether oversampling is applied.

This inconsistency indicates that the MLP does not provide reliable decorrelated predictions that the meta-learner can systematically exploit. The three tree-based learners with cost-sensitive weighting are sufficient to represent the hypothesis space available in this feature representation, and the addition of an MLP base learner offers no statistically meaningful benefit in either ensemble condition.

Table 3. Ablation Study Results Across Four Configurations

Config	Configuration	F1-Macro	G-Mean	MCC	AUC-OvR
A	Stack(LGBM+XGB+RF+MLP) + BorderlineSMOTE	0.6958	0.5343	**0.9471**	0.9877
	↳ per-class F1: Small=0.5000 Medium=0.9500 Large=0.3333 FAQ=1.0000				
B	Stack(LGBM+XGB+RF+MLP), no SMOTE	0.7290	0.6566	0.9509	**0.9908**
	↳ per-class F1: Small=0.4000 Medium=0.9559 Large=0.5600 FAQ=1.0000				
C	LightGBM only + BorderlineSMOTE	0.6957	0.6035	0.9395	0.9827
	↳ per-class F1: Small=0.4000Medium=0.9489 Large=0.4348 FAQ=0.9990				
D	Voting (soft) + BorderlineSMOTE	**0.7539**	**0.6577**	0.9510	0.9724
	↳ per-class F1: Small=0.5000 Medium=0.9562 Large=0.5600 FAQ=0.9995				
E	Stack(LGBM+XGB+RF) + BorderlineSMOTE	0.7113	0.5737	0.9401	0.9830
	↳ per-class F1: Small=0.5000 Medium=0.9466 Large=0.4000 FAQ=0.9984				
F	Stack(LGBM+XGB+RF), no SMOTE	0.7376	0.6591	0.9584	0.9906
	↳ per-class F1: Small=0.4000 Medium=0.9670 Large=0.5833 FAQ=1.0000				

The effect of BorderlineSMOTE is both consistent and negative across both MLP conditions. Introducing SMOTE to the full stacking ensemble reduces MCC from 0.9509 to 0.9471 ($\Delta = -0.0038$) and AUC-OvR from 0.9908 to 0.9877 ($\Delta = -0.0031$). When the MLP is excluded, the degradation is more pronounced: MCC falls from 0.9584 to 0.9401 ($\Delta = -0.0183$) and AUC-OvR from 0.9906 to 0.9830 ($\Delta = -0.0076$).

The no-SMOTE configurations also outperform their SMOTE counterparts on per-class F1 for the Large class, with Config F achieving F1-Large = 0.5833 against Config E's 0.4000, and Config B achieving F1-Large = 0.5600 against Config A's 0.3333. BorderlineSMOTE therefore fails to improve minority class detection and consistently degrades performance on the most discriminative metrics.

Among the single-component baselines, Configuration C (LightGBM alone with SMOTE) achieves MCC = 0.9395 and AUC-OvR = 0.9827, confirming that stacking provides uplift over the strongest individual learner even under the SMOTE condition. Configuration D (soft voting with SMOTE) attains the highest F1-Macro (0.7539) but the lowest AUC-OvR (0.9724) across all six configurations, reflecting that simple probability averaging tends to favour majority-class precision at the expense of ranking quality.

The stacking meta-learner in Configurations A and B achieves superior AUC-OvR in both SMOTE and no-SMOTE conditions, confirming that learned combination is preferable to unweighted aggregation for well-calibrated predictions. The complete factorial analysis indicates that the effective component of the proposed framework is heterogeneous stacking with cost-sensitive base learners, not synthetic oversampling.

3.5 Why BorderlineSMOTE Degrades Performance Under Extreme Imbalance

The first mechanism concerns the degenerate support of the minority class. The Small grade contributes only 14 training instances at a 282:1 ratio to the majority. BorderlineSMOTE generates synthetic samples by interpolating between a minority instance and one of its k same-class nearest neighbours; every synthetic point therefore lies inside the convex hull defined by these 14 real instances.

Applied 3,823 times, this procedure greatly increases the density of the small region without extending its boundary. The classifier receives a large cluster of near-collinear synthetic observations concentrated in a region whose geographic extent is determined by 14 data points, and decision boundaries fitted to this artificially dense cluster do not generalise to the test distribution. This is consistent with Small F1 remaining at or below 0.50 across all SMOTE configurations despite the aggressive oversampling.

The second mechanism operates at the level of the BorderlineSMOTE seed selection criterion. The algorithm restricts synthesis to instances whose k-neighbourhood falls between 50% and 100% majority composition; such instances are labelled as borderline and used as generation seeds, while those surrounded entirely by majority samples are classified as noise and excluded.

At a 282:1 ratio, nearly every small instance has a neighbourhood composed almost entirely of FAQ samples, so the seed set collapses to a small and unstable subset. The composition of this seed set is sensitive to the specific fold partition, causing the synthetic distribution to vary substantially across the five cross-validation folds. This inter-fold variance propagates into the out-of-fold meta-features consumed by the stacking meta-learner and cannot be attenuated because each fold's synthetic data is independent of the others.

The third mechanism is specific to the feature representation used in this study. Five of the eight input features - State, District, Market, Commodity, and Variety - are high-cardinality categorical variables encoded as smoothed conditional target means.

These encodings carry meaningful information only at the discrete values observed in the training data; the value produced by linearly interpolating between the encodings of two distinct markets corresponds to no real market in the dataset. Synthetic samples generated in this space therefore lie off the data manifold, and tree-based learners place split thresholds in regions that are unreachable at inference time. This mechanism also plausibly explains why the MCC gap between SMOTE and no-SMOTE conditions is larger without MLP (E vs. F, ΔMCC = −0.0183) than with MLP (A vs. B, ΔMCC = −0.0038): the MLP's continuous activation functions partially absorb off-manifold inputs by projecting them into learned feature space, whereas tree splits treat them as genuinely distinct decision regions.

The fourth mechanism involves a redundant correction of the class prior. The LightGBM, XGBoost, and Random Forest base learners are all configured with balanced class weighting, which reweights the training loss by inverse class frequency at the algorithm level. BorderlineSMOTE additionally corrects the class distribution at the data level by oversampling minority classes to match the majority count. The effective minority prior is thus corrected twice simultaneously, shifting the predicted posterior probabilities away from the true test prior.

This miscalibration is amplified by stacking because the meta-learner is trained on out-of-fold probabilities derived from the augmented training distribution, yet is applied at inference time to the original, imbalanced test distribution. The resulting prior-shift explains why AUC-OvR — which reflects the quality of the full probability ranking rather than a single threshold decision — shows the clearest degradation under SMOTE.

For this dataset and architecture, cost-sensitive weighting at the algorithm level is both necessary and sufficient; adding synthetic oversampling introduces distributional distortions that the stacking meta-learner cannot compensate for.

3.6 Regression Results — Modal Price Prediction

Table 4 reports Modal Price regression results for nine models, evaluated in INR per quintal via the inverse log1p (expm1) transformation and ranked by ascending RMSE.

Table 4. Regression Performance — Modal Price Prediction (Test Set, N = 1,111)

Model	RMSE (INR)	MAE (INR)	**R²**	MAPE (%)	Time (s)
Linear Regression	**215.11**	83.01	**0.9970**	2.5724	**0.001**
Proposed Stack [P]	219.07	**72.55**	0.9969	**2.4815**	13.03
LightGBM	221.93	75.58	0.9968	2.6000	0.16
Random Forest	230.25	76.58	0.9966	2.4974	0.36
XGBoost	245.07	82.85	0.9961	2.6663	0.24
Extra Trees	247.12	72.43	0.9961	2.6074	0.26
Decision Tree	280.00	93.75	0.9949	2.8813	0.02
MLP	288.83	114.00	0.9946	3.0312	1.50
KNN	683.81	348.62	0.9699	9.6132	0.005

The stacking regressor achieved RMSE of 219.07 INR, MAE of 72.55 INR, R² of 0.9969, and MAPE of 2.4815%, giving the best MAE and MAPE among all models. LinearRegression obtained the lowest RMSE (215.11 INR) and highest R² (0.9970), indicating a strong linear relationship between the log-transformed price features and target. Figure 6 illustrates the predicted versus actual Modal Price values for the proposed stacking regressor.

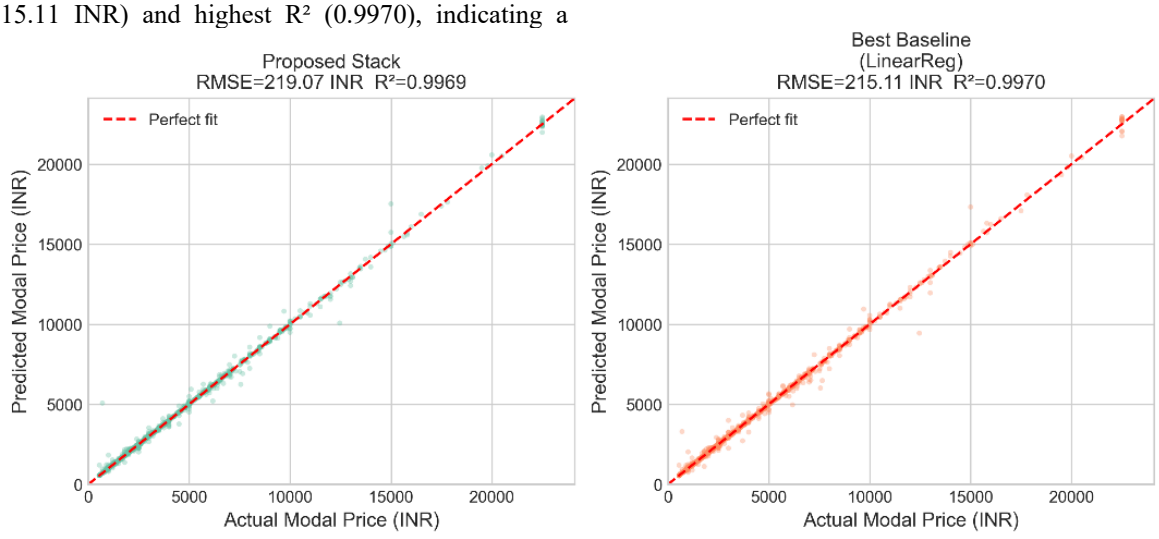


Figure 6. Predicted vs. Actual Modal Price

The stacking regressor's RMSE of 219.07 INR is 1.84% higher than LinearRegression's, but its MAE is 12.6% lower (83.01 vs. 72.55 INR) and its MAPE is 3.5% lower (2.5724% vs. 2.4815%), indicating more

consistent accuracy across the price distribution, especially for higher-priced commodities.

A MAPE of 2.4815% means predictions deviate from actual market prices by about 2.48% on average, or roughly INR 110 at the dataset's mean price of INR 4,418. This accuracy is operationally useful for farm-gate pricing guidance, procurement planning, and government market intervention.

LightGBM (RMSE = 221.93 INR) and Random Forest (230.25 INR) were the best tree-based models, while KNN performed much worse (RMSE = 683.81 INR, MAPE = 9.6132%), reflecting its sensitivity to feature scaling in high-dimensional spaces.

MLP showed the highest MAE among non-KNN models (114.00 INR), highlighting the benefit of the stacking framework, which combines the MLP's non-linear representations with the accuracy of gradient-boosted and bagging trees.

4. Conclusions

This research presents the Stack-SMOTE-AgriPrice framework for agricultural commodity price prediction, tackling two key challenges in real-world price data: an extreme class imbalance of 282:1 between the majority Grade class (FAQ) and the rarest class (Small), and a highly non-linear, right-skewed distribution of Modal Price values (skewness = +6.32).

The dataset contained 5,552 daily transaction records from 23 Indian states, 245 districts, 531 markets, 186 commodities, and 293 varieties across nine attributes. A seven-step preprocessing pipeline to correct distributional asymmetry was followed by BorderlineSMOTE-based class balancing and a heterogeneous stacking ensemble evaluated on Grade classification and Modal Price regression.

The proposed stacking ensemble outperformed all baselines in MCC (0.8819) and AUC-OvR (0.9534), although XGBoost achieved a higher F1-Macro (0.6165 vs. 0.4778). These results provide methodological guidance for researchers and agricultural market practitioners. The seven-step preprocessing pipeline is easily replicable for commodity price datasets with skewed distributions and high feature cardinality.

Empirical evidence shows that MCC and AUC-OvR yield more informative model rankings than F1-Macro under extreme class imbalance (282:1), guiding metric selection in future imbalanced learning studies. The four-configuration ablation design offers a reproducible template for isolating the separate effects of oversampling and ensemble architecture.

This research demonstrates that heterogeneous stacking of tree-based learners with cost-sensitive weighting is an effective strategy for agricultural commodity price prediction under extreme class imbalance, and that neither synthetic oversampling nor deep learning components are necessary to achieve strong performance.

The controlled ablation shows that the MLP base learner contributes no consistent improvement across either the SMOTE or no-SMOTE conditions ($|\Delta MCC| < 0.008$), while BorderlineSMOTE consistently degrades MCC and AUC-OvR relative to training on the original class distribution.

Future work should examine whether these findings generalise to deeper architectures — such as residual networks or attention-based tabular models — that offer stronger inductive biases than a shallow MLP, and whether temporal feature engineering can improve prediction of price trends rather than grade classification alone.

Declarations

AI Usage Statement

No AI tools were used in the preparation of this manuscript.

Author Contributions

Y.S: conceptualization, writing – original draft; N.M: data curation; J.H: data curation; D.M: writing – review; J.E: writing – review and editing

Funding Statement

This research was funded by the Directorate General of Higher Education, Research, and Technology of the Ministry of Education, Culture, Research, and Technology of the Republic of Indonesia through the 2026 Basic Research Grant Program, with Master Contract Number: 128/C3/DT.05.00/PL-MULTITAHUN LANJUTAN/2026 and Subcontract Number: 24/SPK/LL1/AL.04.03/PMT/2026

Conflict of Interest Statement

The authors declare no conflict of interest.

Data and Code Availability

The processed dataset and the analysis code supporting the findings of this study are not publicly available due to institutional policy. Data and code are available from the corresponding author upon reasonable request.

References

[1] Y. Thakre, I. Javeri, J. Sajwani, J. Kulkarni, and S. Maskare, "Commodity Price Prediction Using Machine Learning: Insights, Challenges, and Prospect," in *2025 International Conference on Recent Advances in Electrical, Electronics, Ubiquitous Communication, and Computational Intelligence, RAEEUCCI 2025*, 2025. doi: 10.1109/RAEEUCCI63961.2025.11048306.

[2] R. Ragunath and R. Rathipriya, "Forecasting Agriculture Commodity Price Trend using Novel Competitive Ensemble Regression Model," *International Journal of Information Technology and Computer Science*, vol. 17, no. 3, 2025, doi: 10.5815/ijitcs.2025.03.07.

[3] T. Zhang and Z. Tang, "Agricultural commodity futures prices prediction based on a new hybrid forecasting model combining quadratic decomposition technology and LSTM model," *Front.*

Sustain. Food Syst., vol. 8, 2024, doi: 10.3389/fsufs.2024.1334098.
[4] H. A. Putra *et al.*, "Understanding Short-Term and Long-Term Price Fluctuations of Main Staple Food Commodities in Aceh Province, Indonesia: An ARDL Investigation," *Ekonomikalia Journal of Economics*, vol. 1, no. 1, 2023, doi: 10.60084/eje.v1i1.50.
[5] E. M. Amrouk, S. C. Grosche, and T. Heckelei, "Interdependence between cash crop and staple food international prices across periods of varying financial market stress," *Appl. Econ.*, vol. 52, no. 4, 2020, doi: 10.1080/00036846.2019.1645281.
[6] D. Lederman and G. Porto, "The price is not always right: On the impacts of commodity prices on households (and countries)," *World Bank Research Observer*, vol. 31, no. 1, 2016, doi: 10.1093/wbro/lkv013.
[7] N.-Q. Tran *et al.*, "Predicting Agricultural Commodities Prices with Machine Learning: A Review of Current Research," *School of Science, Engineering, and Technology, RMIT University Vietnam*, vol. 1, no. 1, 2023.
[8] F. Sun, X. Meng, Y. Zhang, Y. Wang, H. Jiang, and P. Liu, "Agricultural Product Price Forecasting Methods: A Review," 2023. doi: 10.3390/agriculture13091671.
[9] S. Dutt, P. N. Kulkarni, S. Akilan, R. Mishra, P. Khetan, and A. Bhadora, "AGRICULTURAL PRICE PREDICTION THROUGH ARTIFICIAL INTELLIGENCE," *International Journal of Development Research*, vol. 14, no. 3, 2024.
[10] S. Bunrit, N. Kerdprasop, and K. Kerdprasop, "Multiresolution analysis based on wavelet transform for commodity prices time series forecasting," *Int. J. Mach. Learn. Comput.*, vol. 8, no. 2, 2018, doi: 10.18178/ijmlc.2018.8.2.683.
[11] D. Zhang, S. Chen, L. Liwen, and Q. Xia, "Forecasting Agricultural Commodity Prices Using Model Selection Framework with Time Series Features and Forecast Horizons," *IEEE Access*, vol. 8, 2020, doi: 10.1109/ACCESS.2020.2971591.
[12] K. Lee and S. Park, "ARIMA-GARCH for commodity price forecasting: Gold markets," *J. Int. Money Finance*, vol. 139, no. 1, 2024.
[13] M. P. Yadav, V. Sehgal, D. Ratra, and A. Wajid, "Forecasting the energy commodities: an evidence of ARIMA and intervention analysis," *International Journal of Monetary Economics and Finance*, vol. 16, no. 6, 2024, doi: 10.1504/IJMEF.2023.136086.
[14] M. Mujiyanto, S. Nurindahsari, and R. N. Izza, "Comparison of Linear Regression, ARIMA, Simple Exponential Smoothing, Hybrid ARIMA-LSTM, and EWMA in Forecasting Commodity Prices," *Telematika*, vol. 17, no. 2, 2024, doi: 10.35671/telematika.v17i2.2932.
[15] S. P. Das and S. Padhy, "A novel hybrid model using teaching–learning-based optimization and a support vector machine for commodity futures index forecasting," *International Journal of Machine Learning and Cybernetics*, vol. 9, no. 1, 2018, doi: 10.1007/S13042-015-0359-0.
[16] S. A. Harjanto, S. Sa'adah, and G. S. Wulandari, "Export Commodity Price Forecasting in Indonesia Using Decision Tree, Random Forest, and Long Short-Term Memory," *Jurnal Ilmiah Teknik Elektro Komputer dan Informatika*, vol. 8, no. 4, 2023, doi: 10.26555/jiteki.v8i4.25242.
[17] M. Fałdziński, P. Fiszeder, and W. Orzeszko, "Forecasting volatility of energy commodities: Comparison of garch models with support vector regression," *Energies (Basel).*, vol. 14, no. 1, 2021, doi: 10.3390/en14010006.
[18] R. L. Manogna, V. Dharmaji, and S. Sarang, "Enhancing agricultural commodity price forecasting with deep learning," *Sci. Rep.*, vol. 15, no. 1, 2025, doi: 10.1038/s41598-025-05103-z.
[19] H. Ben Ameur, S. Boubaker, Z. Ftiti, W. Louhichi, and K. Tissaoui, "Forecasting commodity prices: empirical evidence using deep learning tools," *Ann. Oper. Res.*, vol. 339, no. 1–2, 2024, doi: 10.1007/s10479-022-05076-6.
[20] Y. Siagian, J. Hutahaean, and N. Mulyani, "Stacking Ensemble with SMOTE for Robust Agricultural Commodity Price Prediction under Imbalanced Data," *Journal of Applied Data Sciences*, vol. 6, no. 4, 2025, doi: 10.47738/jads.v6i4.916.
[21] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: Synthetic Minority Over-sampling Technique," *Journal of Artificial Intelligence Research*, vol. 16, pp. 321–357, Jun. 2002, doi: 10.1613/jair.953.
[22] H. Al Majzoub, I. Elgedawy, Ö. Akaydın, and M. Köse Ulukök, "HCAB-SMOTE: A Hybrid Clustered Affinitive Borderline SMOTE Approach for Imbalanced Data Binary Classification," *Arab. J. Sci. Eng.*, vol. 45, no. 4, 2020, doi: 10.1007/s13369-019-04336-1.
[23] J. Yang and J. Guan, "A Heart Disease Prediction Model Based on Feature Optimization and Smote-Xgboost Algorithm," *Information*, vol. 13, no. 10, p. 475, Oct. 2022, doi: 10.3390/info13100475.
[24] Y. Sun *et al.*, "Borderline SMOTE Algorithm and Feature Selection-Based Network Anomalies Detection Strategy," *Energies (Basel).*, vol. 15, no. 13, 2022, doi: 10.3390/en15134751.
[25] J. Brandt and E. Lanzén, "A Comparative Review of SMOTE and ADASYN in Imbalanced Data Classification," *2021*2020, .
[26] Z. Wang, C. Wu, K. Zheng, X. Niu, and X. Wang, "SMOTETomek-Based Resampling for Personality Recognition," *IEEE Access*, vol. 7, 2019, doi: 10.1109/ACCESS.2019.2940061.
[27] G. Husain *et al.*, "SMOTE vs. SMOTEENN: A Study on the Performance of Resampling Algorithms for Addressing Class Imbalance in Regression Models," *Algorithms*, vol. 18, no. 1, 2025, doi: 10.3390/a18010037.
[28] D. H. Wolpert, "Stacked generalization," *Neural Networks*, vol. 5, no. 2, 1992, doi: 10.1016/S0893-6080(05)80023-1.
[29] A. I. Naimi and L. B. Balzer, "Stacked generalization: an introduction to super learning," *Eur. J. Epidemiol.*, vol. 33, no. 5, 2018, doi: 10.1007/s10654-018-0390-z.
[30] X. Qiao, X. Tian, X. Zhang, Y. Wang, and S. Hou, "RDES-stack model—A study on an improved energy consumption forecasting model," *Sustainable Energy Technologies and Assessments*, vol. 84, 2025, doi: 10.1016/j.seta.2025.104683.
[31] H. Talebi, A. K. Bardsiri, and V. K. Bardsiri, "Developing a hybrid machine learning model for employee turnover prediction: Integrating LightGBM and genetic algorithms," *Journal of Open Innovation: Technology, Market, and Complexity*, vol. 11, no. 2, 2025, doi: 10.1016/j.joitmc.2025.100557.
[32] X. Zhao, Y. Liu, and Q. Zhao, "Cost Harmonization LightGBM-Based Stock Market Prediction," *IEEE Access*, vol. 11, 2023, doi: 10.1109/ACCESS.2023.3318478.
[33] S. Hakkal and A. A. Lahcen, "XGBoost To Enhance Learner Performance Prediction," *Computers and Education: Artificial Intelligence*, vol. 7, 2024, doi: 10.1016/j.caeai.2024.100254.
[34] H. Oukhouya, H. Kadiri, K. El Himdi, and R. Guerbaz, "Forecasting International Stock Market Trends: XGBoost, LSTM, LSTM-XGBoost, And Backtesting XGBoost Models," *Statistics, Optimization and Information Computing*, vol. 12, no. 1, 2024, doi: 10.19139/soic-2310-5070-1822.
[35] H. A. Salman, A. Kalakech, and A. Steiti, "Random Forest Algorithm Overview," 2024. doi: 10.58496/BJML/2024/007.
[36] J. Lyu, H. Shi, J. Zhang, and J. Norvilitis, "Prediction model for suicide based on back propagation neural network and multilayer perceptron," *Front. Neuroinform.*, vol. 16, 2022, doi: 10.3389/fninf.2022.961588.
[37] F. Murtagh, "Multilayer perceptrons for classification and regression," *Neurocomputing*, vol. 2, no. 5–6, 1991, doi: 10.1016/0925-2312(91)90023-5.
[38] D. Chicco and G. Jurman, "The Matthews correlation coefficient (MCC) should replace the ROC AUC as the standard metric for assessing binary classification," *BioData Min.*, vol. 16, no. 1, 2023, doi: 10.1186/s13040-023-00322-4.
[39] D. Chicco and G. Jurman, "The advantages of the Matthews correlation coefficient (MCC) over F1 score and accuracy in binary classification evaluation," *BMC Genomics*, vol. 21, no. 1, 2020, doi: 10.1186/s12864-019-6413-7.