Published by :
https://www.ijert.org/
An International Peer-Reviewed Journal

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 11 , November - 2026

SmartSurplus: A Machine Learning-Based Food Surplus Prediction and NGO Redistribution Decision-Support System

Dr. Mrs. Gresha Bhatia, Dr Mrs. Saylee Gharge
Yahya Shaikh, Aman Yadav, Ashutoshkumar Tripathi, Riddhi Menghrajani
Department of Computer Engineering
Vivekanand Education Society's Institute of Technology (VESIT)
University of Mumbai, Mumbai, India

TABLE I
LIST OF ABBREVIATIONS. TERMS USED THROUGHOUT THE PAPER.

Abbreviation	Meaning
NGO	Non-governmental organization
RF	Random Forest
LightGBM	Light Gradient Boosting Machine
MAE	Mean absolute error
RMSE	Root mean squared error
MILP	Mixed-integer linear programming
SHAP	Shapley additive explanations

***Abstract*—Food redistribution requires two decisions: estimating the food that will remain unserved, and selecting a recipient that can receive a confirmed donation within its operating constraints. SmartSurplus addresses these with separate surplus-prediction and NGO-matching models trained on synthetic simulation data. Across ten new surplus populations, full-history LightGBM achieves a mean absolute error (MAE) of 1.894 kg per item-service against 2.044 kg for Random Forest. Across ten matching seeds, LightGBM improves simulated received share over nearest-feasible matching by 5.732 percentage points (95% interval [4.891, 6.587]). A 120-case scheduling benchmark compares greedy and integer-programming assignment under shared capacity. All outcomes are synthetic, and field validation is identified as future work.**

***Index Terms*—unserved food surplus, LightGBM, Random Forest, NGO matching, constrained scheduling, simulation evaluation**

Table I lists the abbreviations used in the paper.

I. INTRODUCTION

Restaurants, canteens and catering services prepare food before demand is fully known. Food that remains unserved may be redistributed, but a surplus estimate alone cannot establish how much food is available for collection or which NGO can receive it in time. UNEP identifies measurement as a foundation for food-waste reduction [1]. Preparation-time forecasting, donation confirmation and recipient selection are therefore kept as distinct decisions.

In SmartSurplus, unserved kilograms are first estimated at service completion. After the provider confirms the quantity and handling requirements, feasible NGO/courier alternatives are ranked using expected receipt and elapsed time. A nearby NGO can be unsuitable because of limited intake capacity, incompatible food or storage, an unavailable courier, or a closing receiving window. Proximity is useful, but it is only one input to the decision.

Three research questions are investigated. First, how much do preparation transformations and completed-service histories contribute to prediction under fixed learner settings? Second, does learned matching improve received mass over equally constrained proximity, time and history/demand rules? Third, what does greedy assignment lose relative to a small integer-program benchmark when offers share NGO capacity and couriers? The contribution of this paper is an auditable empirical evaluation of these linked decisions, rather than a new tree-learning algorithm or an established field deployment.

The main contributions of this paper are as follows:

1) A two-stage decision-support design that separates surplus estimation from NGO matching.
2) A leakage-controlled surplus predictor with a feature-bundle ablation of completed-service history.
3) A feasibility-gated learned ranking of NGO/courier alternatives, compared with proximity, time and history-based rules.
4) A shared-capacity scheduling benchmark comparing greedy and mixed-integer assignment.
5) A frozen, repeated-seed evaluation protocol with paired bootstrap intervals.

The paper first reviews related work and states the problem, then describes the architecture, design and methodology, and closes with results, limitations and future work.

II. RELATED WORK

LightGBM builds an additive tree ensemble using histogram-based boosting [2]; Random Forest aggregates randomized trees [3]. Both can represent conditional interactions between preparation, expected demand, calendar effects and operating history. Specified pipelines and objectives are compared, rather than assuming that either family is universally preferable.

Food-rescue research also addresses the decisions surrounding a forecast. Shi et al. study volunteer recommendation using

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 11 , November - 2026

bandit data-driven optimization and discuss the effects of small and selectively collected historical datasets [4]. Mertzanidis et al. describe Food Drop allocation with explicit efficiency and fairness considerations in a partner deployment [5]. These studies establish that prediction and food allocation already have substantial prior work. The experiments reported here do not reproduce their algorithms, establish superiority over them, or evaluate comparable deployment or fairness guarantees.

Public measured datasets illuminate the data gap. The Finnish school dataset records kitchen/serving waste and plate waste [6], whereas LaDe records last-mile parcel tasks [7]. They are used only in separate related-target benchmarks. Neither provides the aligned preparation-time pure-surplus labels and alternative-NGO receipt outcomes needed to validate the complete system. This scope is central to interpreting the following results.

III. PROBLEM STATEMENT

Two linked decisions are considered. The first is estimating, before service, the quantity of prepared food that will remain unserved for a tracked item-service. The second is selecting, after the provider confirms a donatable quantity, the NGO and courier that maximize the food actually received while respecting compatibility, capacity and time-window constraints. Each decision has its own inputs, outputs and evaluation measure: kilogram error for the first, and received share of offered food for the second. Keeping them separate prevents the surplus estimate from being treated as confirmed donatable food.

IV. SYSTEM ARCHITECTURE

The platform is organized around provider input, prediction, NGO coordination and administrator views. The frontend uses Next.js and the backend uses FastAPI with role-based authentication. Fig. 1 describes how these participants exchange information. The architecture is preserved as a system design, while the experimental implementation is stated separately.

Fig. 1 shows the high-level layout of the platform. Providers submit food and preparation details, the prediction component returns a surplus estimate, and NGO users view and respond to available food. Administrators oversee the process. The matching output appears only after the provider has confirmed what is actually available, which keeps the surplus estimate and the donation offer separate.

Surplus training and evaluation run offline. Saved artifacts contain the feature transformations and fitted models; an inference request does not retrain a model. The enhanced 46-input surplus models in this study are research artifacts and have not replaced the legacy provider prediction flow. The separate NGO Ranking Lab accepts confirmed-offer and candidate snapshots and returns ranked proposals with rejection reasons. The Random Forest component remains a valid comparison path, while the experiments also fit LightGBM. Routing, allocation alerts and impact analytics in Fig. 1 describe intended capabilities. The research ranking endpoint does not reserve capacity, notify recipients or dispatch a donation. The shared-resource scheduler evaluated below is an offline benchmark, not a transactional allocation service.

TABLE II
PREDICTION-TIME INFORMATION AND EXCLUDED OUTCOMES. THE TABLE GROUPS THE MODEL INPUTS BY TYPE AND LISTS THE OUTCOMES THAT ARE WITHHELD FROM THE PREDICTOR.

Group	Available information
Preparation	Prepared kg, expected consumers, portion kg, buffer, price, capacity and planned duration
Context	Provider type, city, meal, item/category, diet, storage, event, calendar and forecast weather
Past services	Completed-service consumption and surplus; provider/city utilization summaries
Excluded	Current actual demand, consumption, weather, donation confirmations and delivery outcomes

V. SYSTEM DESIGN

A. Input Data and Preprocessing

The surplus predictor receives information available after preparation and before service: observed preparation, expected item consumers, menu/context fields, forecast weather and histories from completed services. The original allowlist contains 27 fields. Ten preparation/calendar transformations and nine causal utilization summaries are added, giving 46 inputs in the full model. Table II summarizes their meaning.

Table II has four rows. The Preparation row lists quantities decided by the kitchen before service, such as prepared kilograms, expected consumers, portion size, buffer, price, capacity and planned duration. The Context row lists descriptive attributes of the provider, meal, item, diet, storage, event, calendar and forecast weather. The Past services row lists only information from services that were already completed, namely earlier consumption and surplus together with provider- and city-level utilization summaries. The Excluded row lists outcomes that are never given to the model: actual demand, actual consumption, actual weather, donation confirmations and delivery outcomes. This exclusion prevents information leakage from the future into the prediction.

One-hot category encoding and numeric median imputation with missing indicators are fitted on training data only. Unknown categories are ignored by the encoder. Raw provider/record identifiers and current actual outcomes are excluded. A historical service enters the feature state only if it completed strictly before the current prediction timestamp. During later-period evaluation, a completed earlier test service can update the observed history, but the model weights remain fixed.

B. Surplus Prediction

For each item-service, the target is the unserved surplus, defined as prepared kilograms minus consumed kilograms by service completion, with consumption bounded between zero and the prepared quantity. This target precedes internal reuse, quality rejection, retention and donation confirmation. Unserved surplus is not equated with discarded waste or confirmed donatable food. The quantity later confirmed by the

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 11 , November - 2026

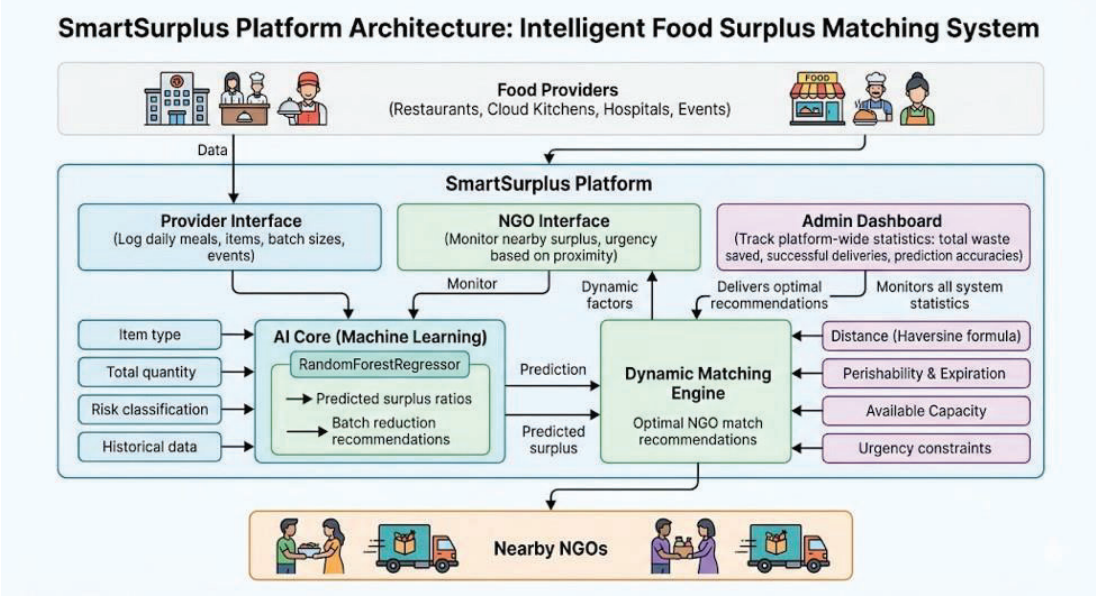


Fig. 1. **System architecture of SmartSurplus.** The diagram connects provider input, surplus estimation and NGO coordination through a common platform. The matching output is reviewed after the provider confirms the available food.

provider is a separate matching input and is never a predictor of surplus.

The selected LightGBM estimates the surplus fraction, which is then multiplied by prepared kilograms after being limited to the range zero to one. Random Forest estimates kilograms directly, and the result is limited to the range zero to the prepared quantity. Both models therefore obey identical physical output bounds. When prepared mass is zero, the output is zero. Prepared mass constrains the result; it does not provide the unknown consumption label.

C. Risk Classification

The interface's surplus-ratio interpretation is retained. The ratio is the predicted surplus divided by prepared kilograms, and is taken as zero when nothing was prepared. Display thresholds are Low for a ratio below 0.20, Medium for a ratio from 0.20 to 0.40, and High for a ratio above 0.40. These are review categories, not probabilities, learned classifiers or biological safety grades. This study evaluates kilogram error and matching outcomes; risk calibration and independently calibrated prediction intervals remain future work.

D. Geographical Analysis

Straight-line proximity between donor and recipient is computed with the haversine great-circle formula, using latitude and longitude in radians and an Earth radius of 6371 km. This gives the shortest surface distance between the two points. Road-distance and travel-time estimates are separate snapshot fields. Simulated routes apply assumed detour and traffic mechanisms; they are not a live routing feed. The nearest-feasible baseline minimizes straight-line distance only after the same feasibility gates used by learned policies.

E. NGO Matching

For each confirmed offer and candidate NGO/courier alternative, a feasibility indicator records whether all known operational checks pass. These include provider confirmation; NGO food/diet/storage compatibility; compatible transport storage; sufficient reported NGO and vehicle capacity in kilograms; report freshness; and pickup, receipt and courier-shift windows. Missing capacity excludes a candidate. The maximum permitted report age is an explicit prototype policy of 120 minutes.

Timing is measured in minutes from the decision. Nominal pickup time is the later of the time the food is ready and the time the courier becomes free plus the approach time. Nominal receipt time is the later of the NGO opening time and the pickup time plus loading, route travel and unloading durations. The gates compare these times with the applicable deadlines and closing or shift times. They cannot eliminate unexpected execution delays.

Random Forest and LightGBM each fit two regressors: one for the received fraction and one for the elapsed receipt minutes. The ranking score of a candidate is its predicted received fraction (limited to the range zero to one) minus a delay penalty, where the penalty is a fixed preference of 0.001 per minute applied to the predicted elapsed time. The candidate with the highest score among feasible alternatives is selected, and no match is made if no candidate is feasible. The expected received kilograms equal the confirmed quantity multiplied by the bounded predicted fraction. The fraction is a bounded regression estimate, not a calibrated acceptance or safety probability. A farther feasible recipient can outrank a nearer one when its predicted receipt benefit is greater.

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 11 , November - 2026

VI. Methodology

This section describes how the experiments were designed and run. It contains six parts: (i) dataset construction, which explains how the surplus and matching datasets were generated; (ii) the creation of nonlinear relationships, which explains the mechanisms built into the simulators; (iii) the preprocessing procedure, which states the data splits, feature controls, frozen protocol, seeds and stress conditions; (iv) the models used, together with their selected configurations; (v) the shared-capacity and courier scheduling benchmark; and (vi) the evaluation metrics and the bootstrap comparison method. Model development, frozen repeated-seed evaluation and shared scheduling are kept separate. Fig. 2 shows how candidate assessment fits the wider coordination workflow. The formal gates and learned score specified above define the evaluated ranking stage; the dispatch step in the diagram does not imply that the application automatically executes a route.

Fig. 2 follows information from its source to the final proposal. Donor and NGO locations and reported capacities enter candidate assessment first. Hard compatibility and time-window gates then remove infeasible candidates, and only the remaining candidates receive a learned score and a rank. The final dispatch step still requires confirmation and is not performed automatically.

A. Dataset Construction

The surplus training dataset contains 60,000 synthetic item-service records: 75 fictional providers, two tracked meal services per day and 400 days, with an 18-item catalog across Mumbai, Pune and Navi Mumbai scenario clusters. One row is a tracked menu item at one provider, meal and date, not an entire kitchen's daily total. A separate 42-day history warmup precedes the published records. All operational quantities, weather and events in this dataset are generated rather than measured. The dataset is used to specify and test mechanisms without claiming a percentage of real-world similarity.

The separately generated matching dataset contains 6,000 confirmed-offer snapshots, ten NGO/courier alternatives per offer and 180 fictional NGO profiles. Its 60,000 rows include 42,000 training, 9,000 validation and 9,000 test alternatives. Whole offers remain in one chronological partition. Only 14,089 eligible training alternatives fit the regressors; 3,028 eligible validation alternatives select configurations. Each alternative has a generated received fraction and elapsed-time outcome, including stochastic delays, partial receipt, rejection and deadline failure. Such potential outcomes for unchosen alternatives are available in simulation only.

Table III summarizes the size and structure of every experiment. The first two rows describe model development: the surplus dataset has 60,000 item-services split chronologically into 42,000 training, 9,000 validation and 9,000 test rows, and the NGO dataset has 60,000 alternatives from 6,000 offers split in the same proportions by offer and time. The remaining rows describe the frozen repeated evaluation. The surplus baseline uses ten new populations of 9,000 final-period services each, giving 90,000 scored rows. The surplus stresses cover four conditions with three seeds each, giving 108,000 scored rows. The NGO evaluation covers ten seeds in each of seven conditions, producing 24,000 offers and 240,000 alternatives. The batch evaluation contains 120 cases, each with ten offers, four NGOs and either one or three couriers. The table therefore distinguishes data used to fit models from data used only for final testing.

TABLE III
Training inputs and frozen repeated-seed evaluation. The table lists each experiment with the number of records and independent units it uses.

Experiment	Records and independent units
Surplus training	60,000 item-services: 42,000 train, 9,000 validation, 9,000 previously viewed test
NGO training	60,000 alternatives: 6,000 offers; 42,000/9,000/9,000 candidate split by offer/time
Surplus baseline	10 new populations; 9,000 final-period services each; 90,000 scored rows
Surplus stresses	4 conditions, 3 seeds each; 108,000 scored rows, including observation-mask reuse
NGO evaluation	10 seeds per 7 conditions; 24,000 offers, 240,000 alternatives
Batch evaluation	10 seeds, 12 cases each: 120 cases; 10 offers, 4 NGOs, 1 or 3 couriers per case

B. Creating Nonlinear Relationships

The surplus generator combines heterogeneous provider scale, seasonal menu/context effects, persistent provider/city demand states, forecast errors, variable portions and occasional disruptions. Preparation responds to expected demand and completed-service history and is rounded into batches. Consumption is constrained by prepared supply, so stockouts produce zero unserved surplus. The resulting interactions and threshold effects need not follow a common straight-line relationship.

Matching combines road detours, traffic, rain, courier queues, changing intake demand and receiving windows. Outcome generation occurs after the known candidate snapshot; actual receipt and realized travel never enter the model inputs. Process stresses modify assumptions before features, gates and outcomes are generated. These mechanisms are explicit assumptions, not parameters estimated from representative field operations.

C. Preprocessing Procedure

For surplus development, the first 280 days supply 42,000 training rows, followed by 60 validation days and 60 test days with 9,000 rows each. Earlier work had already inspected the original test period, so it is treated as descriptive. Selected full models and their hashes were fixed before the repeated evaluation reported here. Three RF and eight LightGBM configurations had been compared on the original validation set; the validation-selected blend assigned zero RF weight and therefore adds no ensemble gain.

Two surplus feature controls are fitted using the selected LightGBM settings: no history, and no history plus seven

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 11 , November - 2026

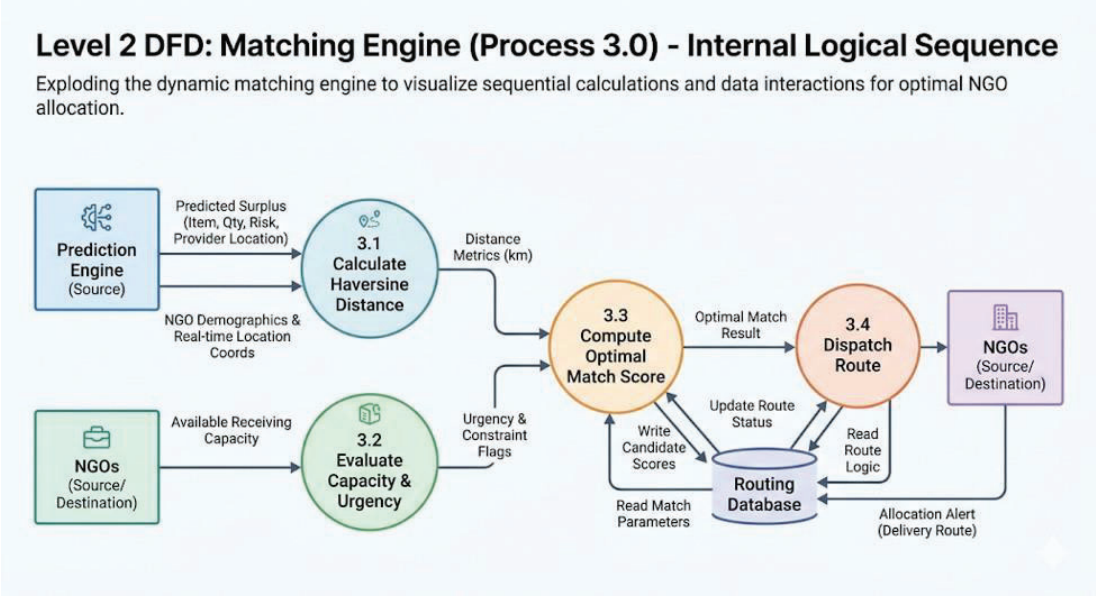


Fig. 2. **Data-flow diagram of candidate assessment and ranking.** The diagram traces location and capacity information into candidate assessment and ranking. In the evaluated matching stage, explicit compatibility and time-window gates precede the learned score; dispatch remains subject to confirmation.

preparation/calendar transformations. Removing history means removing every original history field and every engineered history derivative. The full model restores the whole history bundle and its derived ratios. This is a feature-bundle comparison. A third control removes the four recipient/courier historical-rate/count fields from matching LightGBM while retaining all hard gates.

The frozen protocol, recorded on 20 October 2026 before these new tests, declares surplus baseline seeds 71011–71020, process-stress seeds 71031–71033, NGO seeds 72011–72020 and batch seeds 73011–73020. Each new surplus seed creates an independently drawn provider population with disjoint identifiers. The final 60 days are scored, using preceding completed services for history. Full models are not refitted and app defaults are not chosen from these outcomes.

Surplus stress conditions include a −0.30 log-demand shift from day 340 and a 1.75-fold demand-noise standard deviation throughout generation. Two observation stresses retain underlying baseline outcomes: removal of recorded historical inputs, and independent 40% masking of expected-consumer and weather-forecast fields, with dependent transforms recomputed. Each surplus stress has three seeds. Matching has ten seeds in each of seven conditions: baseline, heavy traffic, tight windows, courier shortage, stale reports, lower NGO demand and missing history. Baseline uses 600 offers per seed and each stress uses 300. Policies within one seed/condition share candidates and potential outcomes; pairing across different conditions is not assumed.

D. Models Used

Two model families are used. Random Forest averages the predictions of many independently grown randomized decision trees. LightGBM builds an additive model in which each new tree corrects the residual error of the trees before it, scaled by a learning rate. Table IV gives the preserved selected configurations.

TABLE IV
FIXED SELECTED MODEL CONFIGURATIONS. THE TABLE LISTS THE HYPERPARAMETERS OF THE FOUR PRESERVED PIPELINES USED IN ALL REPEATED EVALUATIONS.

Pipeline	Settings
Surplus RF	180 trees; depth 14; leaf minimum 8; raw-kg squared-error fit
Surplus LightGBM	600 trees; 15 leaves; leaf minimum 60; rate 0.035; L2 regularization 12; weighted fractional L1 fit
Matching RF	140 trees; depth 14; leaf minimum 20; separate fraction and elapsed-time regressors
Matching LightGBM	300 trees; 15 leaves; leaf minimum 50; rate 0.035; L2 regularization 8; separate fraction/time regressors

For surplus LightGBM, the training target is the surplus fraction (surplus divided by prepared kilograms), and each row is weighted in proportion to its prepared mass for positive prepared quantities. This weighting makes the weighted fractional absolute error identical to the absolute error in kilograms, so the learner optimizes the same quantity that is reported as MAE. The raw-kg Random Forest uses a squared-error objective. The comparison therefore measures two specified pipelines, including objective differences. Matching family configurations were chosen by validation fraction RMSE; the snapshot default was then selected on validation received kg across the family and gated-rule candidates. No linear regression is fitted.

Table IV lists one row per pipeline. The surplus Random Forest uses 180 trees of depth 14 with a minimum leaf size

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 11 , November - 2026

of 8 and predicts kilograms directly under squared error. The surplus LightGBM uses 600 trees with 15 leaves, a minimum leaf size of 60, a learning rate of 0.035 and L2 regularization of 12, and is fitted with a weighted fractional absolute-error objective. The two matching pipelines each train separate regressors for received fraction and elapsed time: the Random Forest uses 140 trees of depth 14, and the LightGBM uses 300 trees with 15 leaves, a learning rate of 0.035 and L2 regularization of 8. These settings were fixed before testing, so differences in results reflect the pipelines and not retuning.

E. Shared Capacity and Courier Scheduling

Snapshot rankings do not consume common resources. A separate set of 120 small batch cases is therefore added: ten seeds, with four cases in each of three conditions (normal, tight NGO capacity and courier shortage) per seed. A case has ten known offers, four NGOs, a common donor hub and three couriers, reduced to one in shortage cases. Departure alternatives are 0, 35, 70 and 105 minutes in a 180-minute horizon. All batch offers are ambient vegetarian grain; these cases do not cover every storage/category interaction in the broader matching dataset.

Each possible action combines an offer, an NGO, a courier and a departure time. The integer program selects the set of actions that maximizes the total predicted received benefit, subject to the following requirements: each offer is assigned at most once; the total full quantity sent to an NGO cannot exceed its capacity; a courier cannot serve two overlapping time intervals; and only actions passing the snapshot feasibility gates may be chosen. Intervals include return to the hub, giving each reused courier a physical starting point. Eligible actions must pass the snapshot gates and return nominally by 180 minutes. Overlap constraints are imposed on the finite intervals between action boundaries. Expected-greedy uses the same predicted benefit and constraints but selects actions greedily. Nearest and earliest greedy policies provide additional controls. This is a discrete batch assignment benchmark, not continuous multi-stop routing or an online arrival policy.

SciPy's mixed-integer solver [9], [10] is used with a 30-second limit and relative-gap target 10^{-6} per solve. A separate non-deployable hindsight solve uses individual potential receipt labels and supplies an upper bound; operational MILP uses predictions only. During replay, each actual job waits for its courier's preceding actual return. Full offered kg remain reserved for the batch, without releasing rejected quantities mid-horizon. The conservative successful-job receipt credit requires pickup, receipt and shift deadlines and return by 180 minutes. A late-return job receives zero credit even if NGO arrival was earlier. This operational metric is distinct from biological spoilage assessment.

F. Evaluation Metrics

Surplus predictions are scored with three measures. Mean absolute error (MAE) is the average absolute difference between observed and predicted surplus in kilograms and reflects typical error. Root mean squared error (RMSE) is the square root of the average squared difference, also in kilograms, and penalizes large errors more heavily. The coefficient of determination (R^2) is dimensionless and compares the squared error of the model with that of predicting the mean; it is zero for a mean predictor and negative when the model is worse. Lower MAE and RMSE are better, but the two errors emphasize different costs.

For a matching policy, simulated received share is the total kilograms actually received divided by the total kilograms offered. Unmatched offers contribute zero to the numerator and remain in the denominator. Batch share uses the successful-job credit defined above. Received share is an allocation outcome, not prediction accuracy.

For method comparison, the paired difference between two methods is computed within each seed and condition. Whole seed values are then resampled 5,000 times, and the 2.5th and 97.5th percentiles of the resampled mean difference give the 95% interval [8]. Positive surplus gains mean reference MAE minus learner MAE; matching gains mean learner share minus reference share in percentage points. Batch cases are aggregated within seed before resampling. These descriptive intervals capture fixed-model variation under the simulators, exclude training/model-selection uncertainty and are not field-performance intervals. Exploratory comparisons are reported without confirmatory significance or multiplicity-adjusted claims.

VII. EXPERIMENTAL RESULTS

A. Surplus Prediction and Feature Controls

Table V reports equal-weight seed means on ten new populations. Full-history LightGBM has the lowest MAE, 1.894 kg, while RF has lower RMSE, 2.964 kg, and higher R^2, 0.219. LightGBM's mean R^2 of 0.137 is modest and limits claims of predictive explanatory power; part of the gap is irreducible demand noise built into the simulator, which bounds the attainable R^2 for any learner. None of these values is a percentage accuracy score.

Table V compares five predictors. The training median, a constant benchmark, has the highest MAE (2.174 kg) and a negative R^2, confirming that a single typical value is a poor predictor. The two LightGBM controls without history (no history, and preparation controls) improve slightly to about 2.07–2.08 kg. Restoring the full history lowers MAE to 1.894 kg, the best value in the table. Random Forest with full history has a higher MAE of 2.044 kg but the lowest RMSE (2.964 kg) and highest R^2 (0.219), which reflects its squared-error training objective.

Fig. 3 presents the same comparison graphically. Each bar corresponds to one model or feature bundle from Table V, and the error bars show the bootstrap variation across the ten populations. The figure makes the size of the feature-bundle effects visible: the gap between the history and no-history models is much larger than the gap between the two no-history variants.

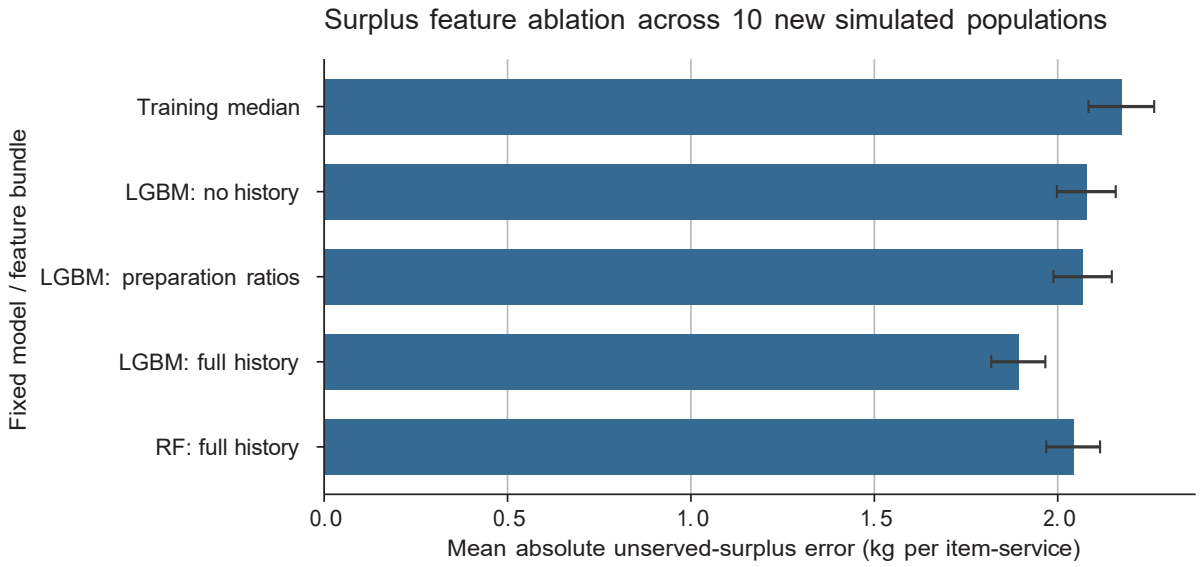
Full-history LightGBM reduces MAE relative to its no-history control by 0.185 kg per item-service, with interval

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 11 , November - 2026

TABLE V
SURPLUS BASELINE RESULTS. EQUAL-WEIGHT MEANS OVER TEN INDEPENDENT SIMULATION POPULATIONS, WITH MAE, ITS 95% BOOTSTRAP INTERVAL, RMSE AND R^2 FOR EACH MODEL OR FEATURE BUNDLE.

Model / feature bundle	MAE (kg)	95% MAE interval	RMSE (kg)	R^2
Training median	2.174	[2.084, 2.263]	3.526	-0.101
LightGBM, no history	2.079	[1.998, 2.158]	3.297	0.034
LightGBM, preparation controls	2.069	[1.988, 2.147]	3.302	0.031
LightGBM, full history	1.894	[1.819, 1.966]	3.115	0.137
RF, full history	2.044	[1.969, 2.115]	2.964	0.219



Source: frozen seed evaluation; 9,000 chronological test services per seed.
Error bars: 95% descriptive bootstrap intervals of seed means; fixed trained models, not field uncertainty.

Fig. 3. **Surplus error comparison across feature bundles.** The plot compares surplus error under fixed selected settings across ten new populations. Each population contributes 9,000 chronological test services. Error bars describe seed-bootstrap variation, not field uncertainty.

[0.174, 0.196]. Against the preparation-control variant the reduction is 0.175 kg [0.164, 0.186], and against full-history RF it is 0.150 kg [0.133, 0.166]. Adding preparation/calendar transforms without history yields a much smaller 0.010 kg reduction [0.003, 0.019].

Table VI reports all declared surplus stresses. Demand shift increases full-history LightGBM MAE to 2.952 kg, and higher noise to 2.259 kg. Each stress has only three seeds. Baseline and stress summaries do not always use the same seed set; the lower missing-forecast mean therefore does not establish that removing information improves prediction. These are condition-specific descriptive results.

Table VI shows how the selected model behaves when conditions change. Removing historical inputs raises MAE modestly from 1.894 to 1.989 kg. The demand shift is the most damaging stress, raising MAE to 2.952 kg, because the model was trained under different demand levels. Higher demand noise gives 2.259 kg. The missing-forecast row shows a slightly lower value of 1.844 kg, but this row uses three seeds against ten for the baseline, so the difference is not evidence that removing forecasts helps. The wide intervals reflect the small number of seeds.

TABLE VI
FULL-HISTORY LIGHTGBM UNDER DECLARED SURPLUS STRESSES. THE TABLE LISTS THE NUMBER OF SEEDS, MEAN MAE AND 95% INTERVAL FOR THE BASELINE AND EACH OF THE FOUR STRESS CONDITIONS.

Condition	Seeds	MAE (kg)	95% interval
Baseline	10	1.894	[1.819, 1.966]
Missing history	3	1.989	[1.917, 2.114]
Missing forecasts	3	1.844	[1.766, 1.979]
Demand shift	3	2.952	[2.879, 3.060]
Higher noise	3	2.259	[2.145, 2.342]

B. NGO Matching and Feature Controls

Table VII summarizes baseline matching. LightGBM receives 58.70% of offered mass on average, compared with 52.97% for nearest-feasible and 58.51% for RF. All gated poli-

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 11 , November - 2026

TABLE VII
NGO BASELINE RECEIVED SHARE ACROSS TEN SEEDS. MEAN PERCENTAGE OF OFFERED FOOD RECEIVED UNDER EACH POLICY, WITH THE 95% BOOTSTRAP INTERVAL.

Policy	Share (%)	95% interval
Nearest feasible	52.97	[51.26, 54.62]
Earliest feasible	53.66	[52.07, 55.20]
History/demand rule	57.40	[55.74, 59.13]
LightGBM, no history	56.89	[54.93, 58.69]
Random Forest	58.51	[56.55, 60.32]
LightGBM	58.70	[56.74, 60.54]

TABLE VIII
PAIRED LIGHTGBM RECEIVED-SHARE ADVANTAGE. DIFFERENCE IN RECEIVED SHARE, IN PERCENTAGE POINTS, BETWEEN LIGHTGBM AND EACH REFERENCE POLICY; POSITIVE VALUES FAVOR LIGHTGBM.

Reference policy	Gain	95% interval
Nearest feasible	5.732	[4.891, 6.587]
Earliest feasible	5.041	[4.283, 5.761]
History/demand rule	1.295	[0.770, 1.774]
LightGBM, no history	1.809	[1.111, 2.589]
Random Forest	0.193	[-0.247, 0.645]

cies select only candidates that satisfy the assumed snapshot constraints; they may still fail during stochastic execution.

Table VII lists six policies in increasing order of received share. The two simple rules, nearest feasible (52.97%) and earliest feasible (53.66%), receive the least. The history/demand rule, which uses recipient acceptance history and demand, reaches 57.40%. LightGBM without history gives 56.89%, showing the value of the history fields, while the full learned policies reach 58.51% (Random Forest) and 58.70% (LightGBM). The overlapping intervals of the top policies indicate that they are close to one another.

Table VIII converts the baseline shares into paired differences computed within each seed. The gain over nearest-feasible is 5.732 percentage points [4.891, 6.587]. The gain over the history/demand rule is smaller, 1.295 points [0.770, 1.774]. Removing prior recipient/courier history reduces LightGBM received share by 1.809 points [1.111, 2.589]. In contrast, the LightGBM–RF interval is [-0.247, 0.645] points; this interval contains zero, so the comparison does not establish a clear family winner.

Fig. 4 reports the paired nearest-feasible comparison in all seven conditions. Gains remain positive over their descriptive intervals, but their sizes differ. Courier shortage gives a 1.426-point gain [1.165, 1.673], while lower NGO demand gives 8.634 points [7.875, 9.558]. These gains are within-condition policy comparisons, not evidence that a stress itself improves outcomes. All 70 seed/condition runs are retained.

Each entry in Fig. 4 shows the received-share gain, with its interval, for one of the seven conditions (baseline, heavy traffic, tight windows, courier shortage, stale reports, lower NGO demand and missing history). A positive value means LightGBM received more food than the nearest-feasible rule under that condition. The figure shows that the learned policy

TABLE IX
MILP VERSUS EXPECTED-GREEDY SCHEDULING. DIFFERENCE IN SUCCESSFUL-JOB RECEIPT SHARE (MILP MINUS EXPECTED-GREEDY), IN PERCENTAGE POINTS, OVER TEN SEEDS FOR EACH BATCH CONDITION.

Batch condition	Gain	95% interval
Normal	15.084	[7.824, 22.679]
Courier shortage	8.438	[3.876, 13.026]
Tight capacity	2.422	[-0.100, 5.270]

is helpful in every condition, with the smallest gain under courier shortage, where few couriers limit the choices available, and the largest under lower NGO demand.

C. Shared Scheduling Results

Table IX reports operational MILP minus expected-greedy receipt-share differences. Normal cases gain 15.084 points and shortage cases gain 8.438 points. The tight-capacity interval crosses zero, so consistent superiority there is not claimed. The expected objective improves in every condition; realized benefit still depends on prediction errors and delayed execution. Fig. 5 separates the known objective from replayed credit.

Table IX has one row for each of the three batch conditions. In normal cases, integer optimization gives the largest gain, 15.084 points, because many feasible combinations exist and greedy choices can block better ones. In courier-shortage cases the gain falls to 8.438 points, since fewer couriers leave less room to improve the assignment. In tight-capacity cases the gain is only 2.422 points and the interval includes zero, so the benefit there is inconclusive.

Fig. 5 contrasts the two assignment methods under the same predicted objective and then shows the credit actually earned when delayed jobs are replayed. The expected objective rises under optimization in every condition, while the replayed credit is smaller and less certain because predictions are imperfect and execution is delayed.

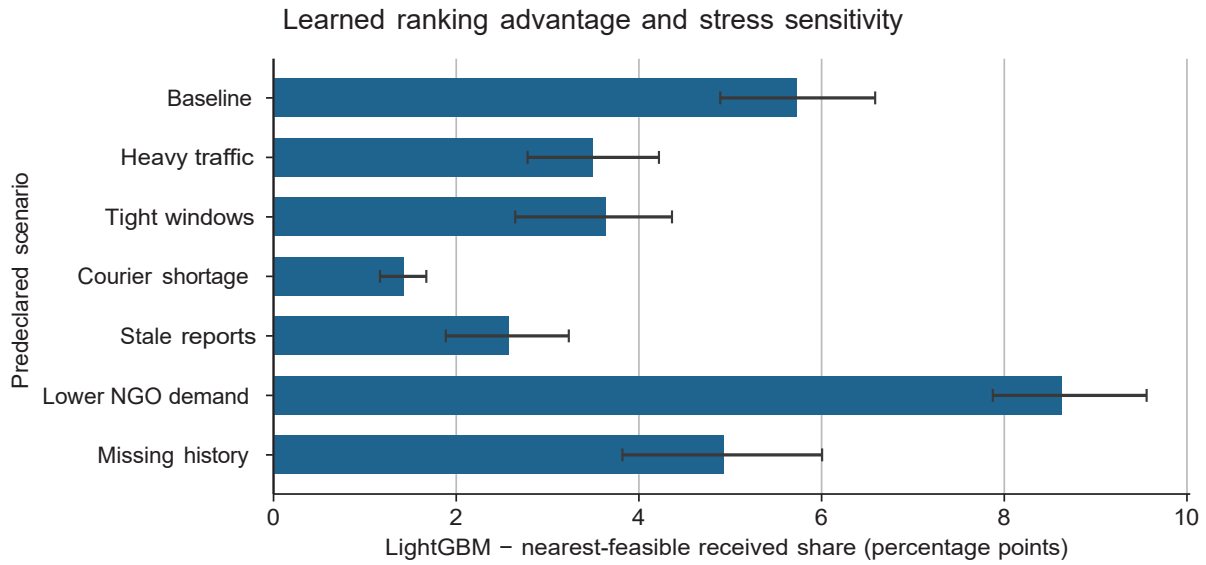
All 120 operational MILP solves terminated successfully. Of 120 separate hindsight solves, 119 terminated successfully and one reached the time limit with a reported relative gap of 0.018953. Thus 239/240 total solves have successful status; the remaining incumbent and bound are retained rather than reporting every solution as an exact optimum. Snapshot gate violations and shared-scheduling capacity/overlap violations are zero in the saved checks. This verifies constraint implementation in assumed cases, not real-world safety or reliability.

D. Related Measured-Data Benchmarks

Models were separately fitted to Finnish school kitchen/serving waste [6]. After 20 history days, the chronological split contains 80 training, 25 validation and 30 test days. The target mixes kitchen and serving waste and lacks prepared mass, so this is not an external test of the 46-input pure-surplus model. LightGBM test MAE is 4.159 kg, versus 5.330 kg for RF and 3.361 kg for the past-five-day mean; both tree R^2 values are negative. The final 20 test days contain the source's intervention period.

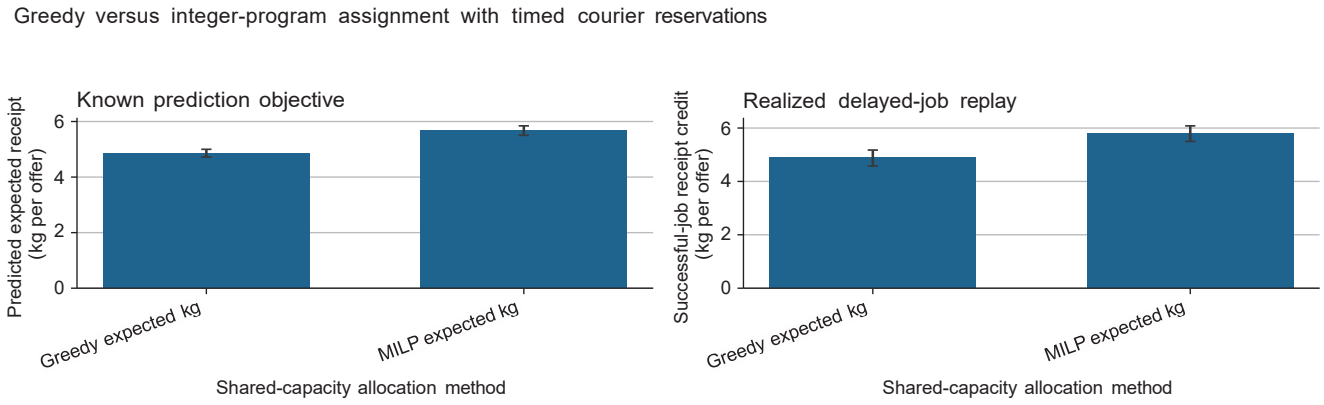
(This work is licensed under a Creative Commons Attribution 4.0 International License.)

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 11 , November - 2026



Source: 10 seeds per condition; 600 baseline offers or 300 stress offers per seed.
Error bars: paired 95% descriptive seed-bootstrap intervals. All outcomes are simulated.

Fig. 4. **LightGBM advantage over nearest-feasible matching in all seven conditions.** LightGBM is compared with the same gated nearest-NGO baseline in every condition. All intervals use paired seed differences within that condition; the scenario populations need not share identical random draws.



Source: 120 synthetic cases, 10 seeds, equal weighting of 3 scenarios. Discrete departures; full offered-kg reservations.
95% descriptive seed-bootstrap intervals; solver status/gaps retained. Replay includes unexpected delays.
Receipt credit requires pickup/receipt/shift windows and full return by 180 min; late-return jobs receive zero credit.

Fig. 5. **Expected-greedy versus integer-optimization batch scheduling.** The same predicted receipt objective is used for both assignment methods. Delayed-job replay is evaluated separately. Successful-job receipt credit also requires return to the donor hub by 180 minutes.

In a separate LaDe parcel experiment [7], 31,357 cleaned records are split by acceptance date into 22,880 training, 7,390 validation and 1,087 test tasks. The target is acceptance-to-completion minutes, including queues and batching. LightGBM MAE is 121.50 minutes, RF 123.45 and the training median 146.21; tree test R^2 values remain negative. These parcel models are not deployed to the food ranker. Neither benchmark establishes NGO receipt quality or an unspoiled-delivery rate.

VIII. MODEL INTERPRETATION AND RISK ANALYSIS

The feature controls provide a limited but useful interpretation: in the specified surplus mechanism, completed-service history contributes more to MAE reduction than the seven added preparation/calendar transforms alone. This is a bundle-level result under fixed learner settings. It does not identify a causal feature effect in restaurants, and the original 27-input schema already contained history before the added controls removed it.

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 11 , November - 2026

For matching, the history ablation improves understanding of policy performance without allowing learned scores to override the gates. Nevertheless, stored acceptance rates may be biased by previously chosen recipients. In actual operation, rejected and unchosen alternatives are not fully observed. Simulation exposes those alternatives, which simplifies evaluation relative to field policy learning.

Surplus-review labels, operational feasibility and biological food safety are distinct. Predicted kilograms cannot certify food quality. Storage declarations, provider confirmation and receipt windows can constrain a proposal, but no temperature-time microbiological model, field rejection study or spoilage calibration was evaluated. The application should therefore support human review of both availability and handling requirements.

IX. DISCUSSION

The two stages favor different conclusions. For typical surplus kilogram error, full-history LightGBM improves on the selected RF and no-history controls across new simulation populations. RF performs better on squared-error-oriented measures. This tradeoff is compatible with their differing training objectives and argues for stating the operational loss before selecting a model.

For NGO matching, the largest defensible result is the improvement over simple, equally gated rules. LightGBM's small observed advantage over RF is uncertain across seeds. Shared scheduling addresses another source of loss: independent good choices can compete for the same NGO intake budget or courier. Integer optimization improves the predicted batch objective, but its realized tight-capacity advantage is inconclusive and its scope remains a small known batch with discrete departures.

The related real-data results also temper the interpretation. A rolling historical mean outperforms both trees on the Finnish test, and parcel elapsed-time errors remain large. These findings are reported as target-specific evidence, not as successful validation of the food workflow. Repeated seeds test sensitivity to random draws within a generator family; they do not validate that family's structural assumptions.

X. OPERATIONAL WORKFLOW

The provider enters a tracked food item, preparation quantity, planned context and preceding observed history. The surplus research pipeline returns a bounded estimate. The provider later checks actual availability and confirms donatable kg and handling windows. For a confirmed offer, the ranking service validates candidate reports, applies compatibility/capacity/time gates and returns ordered proposals with expected received kg, elapsed time and rejection reasons. Receiving capacity, vehicle capacity and offer quantity use kilograms consistently in this experimental interface. A proposal remains distinct from an exclusive acceptance or persisted reservation. The offline batch benchmark keeps reservation ledgers and replayed courier jobs, whereas the current ranking endpoint remains stateless. Connecting those ledgers to persistent transactional application records is required before simultaneous users can rely on them.

XI. PRACTICAL APPLICATION

For a provider, the estimate can support a review of batch size and recent demand. Its unit is a tracked item-service, so it should not be presented as a whole-kitchen daily quantity without a separate aggregation. For an NGO coordinator, ranking can reduce the number of unsuitable candidates to inspect while displaying why others were excluded. A coordinator still needs to confirm recipient capacity, food suitability and transport arrangements.

The evaluation exports predictions, candidate choices, paired summaries, nominal reservations and actual delayed jobs. Independent saved-output verification passed 2,039/2,039 checks. Two focused scheduling tests compare MILP with exhaustive enumeration on a hand-computable fixture and check delayed-job queue propagation. These checks establish reproducibility and implementation consistency, not measured reductions in food waste, live model accuracy or operational deployment readiness.

XII. LIMITATIONS

All pure-surplus and NGO matching labels are synthetic. Coordinates describe scenario areas, not verified premises, and traffic, demand, quality incidents and capacity parameters are assumptions. New identifiers and seeds produce new draws within the same structural family; they are not independent field distributions. The two training datasets support separate stages rather than one validated end-to-end production outcome.

Full models were fixed for repeated testing, so the intervals exclude retraining and selection uncertainty. Each surplus stress has only three seeds, and multiple exploratory comparisons receive no confirmatory multiplicity correction. Snapshot policies do not share capacity across offers; the separate batch experiment addresses shared resources only for small, ambient-food, common-hub cases. It excludes online arrivals, split offers, multi-stop routing, dynamic capacity releases and fairness objectives.

The enhanced surplus model is not yet the main provider flow, the NGO lab produces proposals only, and the offline scheduling ledger has no transactional production integration. No compatible measured pure-surplus or alternative-NGO receipt test was obtained. The Finnish and parcel targets differ materially. No biological spoilage assurance, risk-probability calibration or verified field impact follows from this study.

XIII. FUTURE WORK

The next empirical step is a timestamped provider dataset containing planned inputs, prepared kg and separately weighed unserved kg, followed by confirmed donation quantities and actual receipt/rejection times. It should preserve prediction-time provenance and include unseen providers and operating

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 11 , November - 2026

periods. Recipient and courier reports should retain age, capacity units and later corrections so that stale information and selection bias can be studied.

For the platform, persistent exclusive reservations, recipient acceptance, courier availability updates and outcome feedback are planned. For scheduling, extensions include rolling arrivals, route-network travel, dynamic releases and multiple food/storage types. Risk and interval calibration require separate validation; biological safety would require appropriate measured handling and temperature information. Any updated selection criterion should be fixed before a new field test is inspected.

XIV. CONCLUSION

SmartSurplus is studied as two linked decisions: estimating unserved food before service and ranking feasible recipients after donation confirmation. Under a frozen repeated-seed simulation protocol, full-history LightGBM achieves surplus MAE 1.894 kg compared with RF's 2.044 kg, while RF retains lower RMSE and higher R^2. Learned matching improves on nearest-feasible allocation by 5.732 percentage points, but LightGBM and RF are not clearly separated. A shared-capacity scheduling benchmark shows where optimization can improve on greedy choices and where realized gains remain uncertain. The architecture and data-flow diagrams (Figs. 1 and 2) explain how these decisions fit the platform. Reproducible artifacts and explicit constraints strengthen the experimental account. Modest explanatory power, structural simulation assumptions and target-mismatched real benchmarks leave field accuracy, delivery benefit and food safety unresolved.

ACKNOWLEDGMENT

The authors thank their mentor, Dr. Mrs. Gresha Bhatia, Dr Mrs. Saylee Gharge for guidance throughout this work. AI-based tools were used for code review and language editing; the authors take full responsibility for the content.

REFERENCES

[1] United Nations Environment Programme, *Food Waste Index Report 2024*, 2024. https://www.unep.org/resources/publication/food-waste-index-report-2024
[2] G. Ke et al., "LightGBM: A Highly Efficient Gradient Boosting Decision Tree," in *Advances in Neural Information Processing Systems*, vol. 30, 2017. https://proceedings.neurips.cc/paper/2017/hash/6449f44a102fde848669bdd9eb6b76fa-Abstract.html
[3] L. Breiman, "Random Forests," *Machine Learning*, vol. 45, pp. 5–32, 2001, doi:10.1023/A:1010933404324.
[4] Z. R. Shi, Z. S. Wu, R. Ghani and F. Fang, "Bandit Data-Driven Optimization for Crowdsourcing Food Rescue Platforms," in *Proc. AAAI*, vol. 36, no. 11, pp. 12154–12162, 2022, doi:10.1609/aaai.v36i11.21475.
[5] M. Mertzanidis, A. Psomas and P. Verma, "Automating Food Drop: The Power of Two Choices for Dynamic and Fair Food Allocation," arXiv:2406.06363, 2024. https://arxiv.org/abs/2406.06363
[6] A. Kuusisto, J. Latva and E. Wilén, "Food waste dataset: secondary school in Finland 2024–2025," Zenodo, 2026, doi:10.5281/zenodo.20825233.
[7] L. Wu et al., "LaDe: The First Comprehensive Last-mile Express Dataset from Industry," in *Proc. KDD*, pp. 5991–6002, 2024, doi:10.1145/3637528.3671548.
[8] B. Efron, "Bootstrap Methods: Another Look at the Jackknife," *Ann. Statist.*, vol. 7, no. 1, pp. 1–26, 1979.
[9] P. Virtanen et al., "SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python," *Nature Methods*, vol. 17, pp. 261–272, 2020.
[10] Q. Huangfu and J. A. J. Hall, "Parallelizing the Dual Revised Simplex Method," *Math. Program. Comput.*, vol. 10, pp. 119–142, 2018.

(This work is licensed under a Creative Commons Attribution 4.0 International License.)