IOSR Journal of Computer Engineering (IOSR-JCE)
e-ISSN: 2278-0661, p-ISSN: 2278-8727, Volume 28, Issue 3, Ser. III (May. – June. 2026), PP 63-77
www.iosrjournals.org

Multimodal Explainable Learning for Joint Crop-Yield and Price Forecasting

Vivek Kumar*
Department of Computer Science and Engineering, JSPM University,
Pune – 412207, Maharashtra, India
ORCID iD: https://orcid.org/0009-0009-4839-2174
**Corresponding Author*

Rahul R. Chakre
Department of Computer Science and Engineering, JSPM University,
Pune – 412207, Maharashtra, India
ORCID iD: https://orcid.org/0000-0002-2840-2007

Abstract: *The forecasting of crop yield and of market price is needed for procurement planning and price-risk management, yet the two are predicted separately, though shared weather and crop conditions drive both. Here the two are forecast together by a single model, a Modified CNN-LSTM with multi-head attention, demonstrated on wheat in Haryana, India. Four public sources are drawn on, namely the crop-yield statistics of DES; daily AGMARKNET prices; the NDVI, EVI, and NDWI indices from Sentinel-2 and MODIS via Google Earth Engine; and ERA5-Land weather data, aligned to a district-week grid. In early fusion, the features are encoded together; in late fusion, yield is predicted first, and its supply signals are passed to a separate price model. On the held-out 2025 season across 22 districts, yield yields an R^2 of 0.762 and a MAPE of 4.13% under late fusion, and price an R^2 of 0.821 and a MAPE of 4.34% under early fusion, outperforming XGBoost and Random Forest. Each forecast includes a bootstrapped 90% prediction interval and attributions from Integrated Gradients and LIME. Since every stage is parameterised by the crop calendar, satellite identifiers, and market codes, the pipeline extends to crops and regions where public data are available.*

Keywords: *Crop yield forecasting, agricultural price prediction, multimodal fusion, CNN-LSTM, explainable AI, wheat.*

Date of Submission: 08-06-2026 Date of Acceptance: 19-06-2026

I. INTRODUCTION

Reliable forecasts of how much a crop will produce, and of what it will sell for, are needed wherever agricultural produce has to be procured, stored, and transported in large quantities. The case taken up in this paper shows why this matters. Haryana, despite the fact that it accounts for less than 2% of India's total land area, produces about 11% of the country's wheat. To obtain that much from such a small area, careful planning is needed for everything that happens with the wheat. The government agencies that buy the wheat book space in warehouses many months before farmers bring it in in April and May. As early as February, traders at the agricultural markets (called mandis) start to give a rough idea of prices based on how the crop is doing. It can often be observed whether the crop is having problems from pictures taken by satellites. For this reason, a good and sensible prediction of both how much wheat there will be and what the price will be is not made for its own sake. These figures are used for warehouse agreements, for ways to reduce risk, and for all the transporting of the wheat from the farm to the people buying it.

The usual tools available today rarely give that sensible agreement. One set of programs analyses satellite images and weather data to estimate the harvest, while price predictions come from completely separate programs that mostly look at past mandi behaviour. The two do not 'talk' to each other. Therefore, when their predictions diverge around harvest time, which they often do because the crop's satellite-observed health is not yet reflected in price expectations, the differences have to be reconciled by hand, which makes each set of predictions less accurate.

Machine learning has improved each prediction individually. For figuring out the harvest amount, 1D-CNN and LSTM models that make use of NDVI (a measure of plant greenness) and ERA5 weather data are as accurate as complicated computer simulations of the crop for each individual district. For predicting prices, LSTMs and Transformer-based models with weather details have had lower average percentage errors than older

ARIMA-type models. The individual predictions are not the problem. The problem is that the two are kept separate and do not share their reasoning, their confidence, or their inputs. As a result, a model expecting a smaller harvest produces no matching rise in the price it forecasts.

This research addresses this problem. In this paper a general framework is described that predicts both the harvest of a crop and its market price, using a single 'thinking' section and only information available to everyone. The framework is developed and validated on wheat and its mandi price in Haryana, although nothing in its design is tied to either. Section II examines other work on this. Sections III, IV, and V identify areas where more information is needed, state the problem, and outline what the study aims to achieve. Section VI gives all the details of the methodology. Section VII describes the information used. Section VIII shows the results of the experiments. Sections IX, X, and XI cover explainability, discuss what the results mean, and then conclude the research.

II. LITERATURE SURVEY

A large number of papers published between 2015 and 2025 were reviewed, and the ones most directly relevant to this work were selected. Table 1 lists them. They fall into three broad areas. The first is using remote sensing and machine learning to predict how much crop a region will produce. The second is using deep learning to forecast agricultural commodity prices. The third, of which there are not many, is trying to do both things at once within a single system.

Table 1. Relevant prior work

Name of the Author	Qualitative Analysis	Quantitative Analysis
Manogna et al.[1]	Compared LSTM and GRU on noisy Indian commodity prices; both outperformed classical models by a wide margin.	R^2 improved by 0.10 to 0.25 depending on the commodity; RMSE fell by as much as 75 per cent in the best case.
Min et al.[2]	Compared graph neural networks with recurrent models for international agricultural commodity price forecasting.	The graph-based model showed lower RMSE than the recurrent baselines on most commodities tested.
Nayak et al.[3]	Meta-Transformer with metaheuristic tuning, tested at multiple forecast horizons on agricultural price data.	MAPE dropped from roughly 12 per cent down to about 9 per cent.
Ashfaq et al.[4]	ML regression models for wheat yield using climate variables and NDVI.	R^2 showed a clear improvement over simpler approaches, although the paper did not disclose exact test-set numbers.
Nayak et al.[6]	LSTM-based price forecaster augmented with rainfall, temperature, and lagged prices for Indian commodities.	RMSE was 15 to 30 per cent lower than ARIMAX when both were evaluated on the same test splits.
Pandit et al.[7]	Two-stage ARIMAX → LSTM hybrid: ARIMAX captures linear seasonal yield patterns, LSTM learns the nonlinear residuals.	Achieved an R^2 of roughly 0.93, with RMSE dropping by 10 to 18 per cent compared to using either ARIMAX or LSTM on its own.
Wolanin et al.[9]	1D-CNN on daily weather for the Indian wheat belt; activation maps used for post-hoc temporal explanation.	Brought MAE down to about 7 per cent from roughly 12 per cent for the baseline, with an R^2 of around 0.8.
Maimaitijiang et al.[10]	Fused UAV RGB and multispectral imagery in a deep network for soybean yield; neither sensor alone matched the fused model.	Reached an R^2 of 0.94, with prediction error roughly cut in half compared to using a single sensor.
Sun et al.[11]	CNN–LSTM pipeline: CNN extracts spatial features from MODIS NDVI and climate grids, LSTM captures temporal patterns. US county-level soybean yield.	Achieved an R^2 of 0.9.
Najjar et al.[13]	LSTM on multi-temporal satellite imagery for subfield yield prediction (soybean, wheat, rapeseed). Feature-attribution methods confirmed the model learned agronomically meaningful signals aligned with crop phenology.	Adding more modalities and satellite instances increased cross-validation R^2; attribution analysis identified tillering and flowering as key wheat growth stages.

Two patterns can be observed across the table. First, deep learning models, namely CNNs, LSTMs, Transformers, or their combinations, consistently outperform older approaches such as ARIMA or standard regression. The margin varies by study, although the direction does not [1, 7, 8]. Combining input types also helps. On the yield side, models that use both vegetation indices and weather variables perform better than those that rely on either one alone [4, 9, 10]. On the price side, the models that perform best are those that use weather, supply indicators, and past prices together rather than any single stream [1, 5, 6]. More recent work has begun to open up the yield models, using SHAP and related attribution methods to trace which inputs and which growth stages a prediction rests on [13], [14], and a small number of district-scale Indian studies find the same climate-to-yield relationships holding locally [15], [16]. What is rarely seen is anyone joining the two sides. Those who model yield seldom ask what their numbers imply for the mandi, and those who model price rarely feed in live

crop-condition data, since they make use of last year's harvest figure instead of what the satellite shows today. For the most part the two literatures have been developed in parallel.

III. RESEARCH GAPS

- Almost without exception, yield models and price models are built independently of each other. Satellite-based price studies use past harvest records rather than the crop's current condition. The reverse is also true. Whenever crop-condition variables do appear in a price model, they appear as stale, lagged values rather than a live supply signal [1], [2], [5], [6].
- Most exogenous price models condition on past climate, but they rarely draw on present-day vegetation status from satellite observations. This is despite the fact that satellite-derived indices are the freshest and highest-resolution indicator of available supply at the district level [5], [6], [23].
- There is a basic but overlooked problem. Yield data is published once per season, while mandi prices shift week to week. A useful joint forecast needs to bridge that gap, with annual supply on one end and weekly price on the other, without breaking the cause-and-effect chain that connects them. As far as can be determined from the literature, this has not yet been done [1], [3], [5], [6].
- When a procurement officer is told that "wheat yield will be 4,200 kg/ha", the question that follows is how sure the estimate is. Most models in the literature have no answer to this. They output a single number and stop there. Without a confidence range or a shortfall probability attached, that number does not really help someone who has to sign a purchase contract or decide how large a buffer stock to maintain [6], [8], [12], [17].
- Real-world messiness, such as clouds blocking satellite passes, price gaps on market holidays, and whole districts with missing records, is dealt with differently, or not at all, from one study to the next. Very few of these systems have actually been shown to hold up when fed live, current-season data rather than a clean retrospective dataset [1], [2], [3], [6].

IV. PROBLEM STATEMENT

It is well known that crop yield and market price respond to the same underlying agro-climatic conditions; yet, in practice, they are almost always forecast in isolation from each other. Consider that a bad spell of weather which thins the harvest is the very same event that, weeks later, tightens supply at the mandi and pushes prices up. Despite the fact that this connection is well understood, very few systems attempt to model both quantities together. Here we build a single, explainable forecasting framework for doing so, and we demonstrate it on wheat yield and its corresponding mandi price across the 22 districts of Haryana. The framework draws on weather reanalysis and satellite crop-growth signals alongside market and economic indicators, because we believe that supply and demand planning benefits from treating the two sides as one problem rather than handling them separately.

V. RESEARCH OBJECTIVES

1. To develop a unified model, along with a single pipeline around it, for forecasting both district-level crop yield and weekly market price, so that the two estimates come from the same source and agree with each other rather than needing to be reconciled afterwards.
2. To combine three data streams that are usually studied separately, namely weather variables, satellite-derived crop-condition indicators, and market signals, into one joint representation that captures how crop conditions and market behaviour influence each other, including delayed effects such as a spell of February heat stress showing up as a mandi price spike several weeks later.
3. To keep the joint forecasting transparent, so that a farmer, a trader, or a policymaker can see why a given yield or price came out the way it did and act on it with some confidence rather than taking it on trust alone.

VI. RESEARCH METHODOLOGY

As depicted in Figure 1, the system consists of six stages, where output from each stage feeds the next stage. It ingests four publicly available repositories: remote sensing images, weather reanalyses, markets, and agricultural statistics like Sentinel-2, MODIS, ERA5-Land, Agmarknet, and ICAR statistics. The data set is preprocessed and transformed to the grid level (district-week) – (22 districts, weekly). The system generates three feature groups: crop status (NDVI), weather stress (temperature, humidity, rainfall deficit), and market behavior (price lags, volume trends). Two architectures ingest these three feature types: Early Fusion (by concatenating three features into one input encoder) and Late Fusion (predicts yield, and generates signals). They predict yields and prices at 90% prediction intervals. The interpretability component follows all models to facilitate explanations. The framework integrates the harvest size and its prices as coupled processes and predicts both simultaneously instead of individually.

Timeliness is essential in this context, since only lagged variables, namely prices, rainfall, and satellite indices that are already known at the time of forecasting, enter the models. The overall data flow in our pipeline proceeds through data ingestion (6.1), feature engineering (6.2), the fusion architectures (6.3-6.4), baseline models (6.5), explainability (6.6), and mathematical formulation (6.7). Although the examples in this paper focus on Haryana wheat, the parameter settings can be tuned for application to other crops and regions.

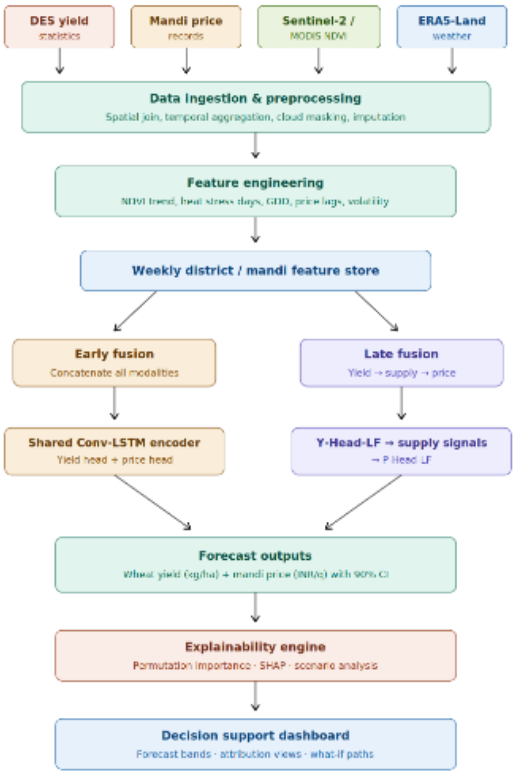

Fig. 1. Full pipeline for joint wheat-yield and mandi-price forecasting.

We begin with four raw data streams, namely satellite imagery, weather grids, mandi transactions, and crop statistics, which are cleaned and aligned to a common district-week grid before being organised into three feature groups. These features then follow two parallel routes. In early fusion, the groups are concatenated into a single CNN-LSTM encoder with one head for yield and another for price. In late fusion, a standalone yield model runs first and passes its supply-side signals (predicted yield, yield anomaly, supply index, shortfall flag) into a separate price model. Both routes output point estimates with 90 % prediction intervals. An explainability module accompanies every model and every individual prediction.

6.1. Data Ingestion and Preprocessing

The study region is broken down into administrative districts at a GAUL level-2 designation, and every single fact in this research is linked to one of those; for the case study this means the 22 districts of Haryana. The four types of information are each cleaned separately, before they are combined. With the mandi (market) records, which arrive daily, the most commonly found price for each district is determined for each week, and total up how much product came into the market during that week. Satellite images from Sentinel-2 and MODIS are first cleared of clouds (using QA60 for Sentinel-2 and SummaryQA for MODIS) and then, for each area that represents a district, the average light reflectance is calculated. If a district is not covered by a useful satellite image in a particular week, the last useful number available is carried forward. Care is taken not to use future data, because if tomorrow's NDVI were included, the model would appear to perform better than it actually is. ERA5-Land, by contrast, is abundant. Its hourly figures are averaged to daily values per district and then summarised weekly. Markets aren't open every day, and sometimes have a gap of one or two days in their prices; these gaps are filled with the most frequent price from the previous seven days for that same market. Finally, as a last tidy up, if a price is more than four Interquartile Ranges above the typical price for its district and month, it is treated as an unusual value and replaced with a smoother price, averaged over the four weeks before. After all that preparation, the

result is a nice, complete set of weekly information for 22 districts and 24 Rabi (winter) seasons, from 2001 to 2025 [22], [23].

6.2. Feature Engineering

Once the weekly panel is ready, the available information is sorted into three groups based on which aspect of the problem each one addresses. The first group addresses the condition of the crop itself and is built entirely from satellite observations. This group calculates the average NDVI, EVI, and NDWI values for each district during the main growing window, as well as the change in NDVI from one week to the next. On top of that, heat-stress days are counted, defined as any day when the maximum temperature exceeds 30°C, starting the count from 1 November each year; both the start date and the temperature threshold are crop-calendar parameters that would be reset for a different crop or region. Growing-degree days are calculated using a base of 5°C. There is also a simple yes-or-no stress indicator that gets switched on whenever a district's NDVI in a particular week falls below the 10th percentile of what that same district has historically shown during that calendar week, which provides a way of spotting weeks where vegetation health is unusually poor [4], [9]. The second group concerns weather-related stress and is derived from the ERA5-Land climate data. It includes the weekly average, minimum, and maximum air temperatures measured at 2 metres above the ground, along with the average relative humidity and wind speed. A rolling 12-week rainfall shortfall is also computed, measured against the 20-year average for that district and time period, because a deficit that builds up over two or three months matters more for wheat than a single dry week does [22]. The third group captures what is going on in the market itself. This group includes the most common prices from 1, 2, 4, and 8 weeks earlier, so the model has a sense of where prices have been heading. Current weekly arrivals are divided by the average over the previous four weeks, yielding a ratio above 1 when more grain than usual is arriving and below 1 otherwise. The standard deviation of the four-week logarithmic returns is also computed as a measure of price volatility. Lastly, simple flags are added to mark weeks that fall within the government procurement season and those affected by market holidays, since both tend to distort normal trading patterns.

6.3. Early Fusion (EF) Architecture

The idea behind Early Fusion is straightforward. Every feature that belongs to a particular district and week is combined into a single input vector. This combined vector is then passed through two one-dimensional convolutional layers, each with 64 filters and a kernel size of 3, using ReLU as the activation function. These layers pick up short-term patterns, such as a two-week NDVI dip signalling sudden crop stress, or an unexpected rise in grain arrivals. Their output then goes into a two-layer Bidirectional LSTM with 128 units per direction and a temporal attention mechanism, which captures dependencies across the entire growing season that convolutional layers alone cannot capture. From the encoder's output, two separate prediction heads branch off. One head is for yield, going through a Dense layer with 64 neurons, then a Dense layer with 1 neuron, producing an estimate along with a Gaussian distribution. The price head has the same structure. For the plain CNN-LSTM baseline, both heads are trained jointly on a variance-normalised sum of the negative log-likelihoods for both targets. The Modified CNN-LSTM replaces this with the weighted MSE and Huber loss defined in Section 6.7.1. For the optimiser, Adam is used with a learning rate of 0.001, and ReduceLROnPlateau is applied, scheduling together with early stopping that has a patience of 15 epochs [11].

6.4. Late Fusion (LF) Architecture

Late Fusion works in the opposite direction compared to Early Fusion. Instead of mixing all the features at the start, a yield model is first trained on its own, using only the supply-side information, namely the satellite indices, weather variables, and crop statistics, and running it through the same convolutional-LSTM structure described above. Once this yield model is ready, three pieces of information can be extracted from it and passed on to the price side. The first is the predicted yield, expressed as a percentage deviation from the five-year average for that district, indicating whether the coming harvest is above or below normal. The second is termed a supply surprise, and it is simply a measure of how much the yield forecast has changed over the previous four weeks. If a prolonged heat spell has been worsening and the satellite-derived NDVI has continued to fall, this number will be large; if conditions have stabilised, it will be close to zero. The third signal is the shortfall probability, which is computed by running a 50-iteration bootstrap ensemble and counting how many of those runs predict a yield below 85 per cent of the five-year district mean. These three numbers, together with the market-behaviour features from the third feature group, are then fed into a separate price model that uses the same convolutional-LSTM architecture. The main advantage of splitting it into two stages like this is that it becomes possible to trace price movements back to a particular supply-side signal, rather than having them emerge from a feature interaction within a single network that would be difficult to interpret [17].

6.5. Baseline Models

We evaluate the modified CNN-LSTM against four baselines. The first two are tree-based, namely XGBoost trained with 500 boosting rounds, a maximum tree depth of 6, a learning rate of 0.05, and 80 per cent row subsampling per round, and Random Forest with 500 trees and a maximum depth of 12. The third is a plain CNN-LSTM built on the same early fusion encoder described in Section 6.3 but without the multi-scale convolutions, the Squeeze-and-Excitation block, or the PriceBoost price head. The fourth is the Temporal Fusion Transformer, run with its standard published configuration. All four baselines are trained on the same feature table and the same train–test splits as the Modified CNN-LSTM, with no derived features added and evaluation windows that are identical. Section VIII gives the full results.

6.6. Explainability and Uncertainty

Explainability in our framework is handled at two levels, one concerning the model as a whole and the other concerning each individual prediction it makes. At the global level, integrated-gradient attributions are averaged across all test samples, and the result indicates which input channels and timesteps the model tends to rely on overall. At the local level, for a specific forecast such as the yield prediction for Karnal district in the second week of February 2023, gradient backpropagation works out how much each input feature pushed that prediction away from the dataset average, which answers what it was about that district-week that made the model expect a higher or lower yield than usual. For uncertainty, a single point estimate is not produced and left at that. Instead, a bootstrap ensemble is run consisting of 50 separate training runs, each on a slightly different resample of the data, and the 50 resulting predictions are collected. The interval between the 5th and 95th percentiles of those predictions forms the 90 per cent prediction interval. It was noticed during development that these intervals were slightly too narrow in some parts of the distribution, so a calibration step is applied on the validation set: if the stated 90 per cent coverage is not actually being met, the intervals are widened until it is, and similarly narrowed if they are too conservative. A scenario-analysis component lets a user take a baseline forecast and vary inputs, for example 20 per cent more heat-stress days than recorded, or a 200 rupee per quintal rise in the minimum support price, and re-run the model to see the new predicted distribution alongside the original, which reveals which inputs the forecast reacts to most strongly [9], [17].

6.7. Mathematical Formulations

This section presents the mathematical details for the two model families that perform best on the held-out test set, namely the Modified CNN-LSTM (which performs best on both targets) and its PriceBoost price head, as well as the two fusion strategies that define the pipeline. The XGBoost, Random Forest, and Temporal Fusion Transformer baselines follow their standard published formulations and are not reproduced here for brevity. Equations are numbered continuously.

6.7.1. CNN-LSTM Modified

The Modified CNN-LSTM extracts multi-scale temporal patterns through parallel 1-D convolutions of kernel sizes 3 and 5:

$$h_3 = ReLU(Conv1D_{k=3}(X)), \quad h_5 = ReLU(Conv1D_{k=5}(X)) \tag{1}$$

$$h_{conv} = [h_3 \,\|\, h_5] \in \mathbb{R}^{T \times 2F} \tag{2}$$

A Squeeze-and-Excitation block applies channel attention:

$$z = GAP(h_{conv}); \quad s = \sigma(W_2 \cdot ReLU(W_1 \cdot z)); \quad \hat{h} = s \odot h_{conv} \tag{3}$$

Sinusoidal positional encoding is then added before the BiLSTM:

$$PE_{t,\,2i} = \sin(t / 10000^{2i/d}), \quad PE_{t,\,2i+1} = \cos(t / 10000^{2i/d}) \tag{4}$$

$$h_t^{bi} = BiLSTM(\hat{h} + PE) \in \mathbb{R}^{2 \times 128} \tag{5}$$

Temporal attention produces a context vector summarising the entire sequence:

$$\alpha_t = \exp(w^T h_t^{bi}) / \Sigma_\tau \exp(w^T h_\tau^{bi}), \quad c = \Sigma_t \alpha_t h_t^{bi} \tag{6}$$

The PriceBoost price head consists of three residual blocks with LayerNorm:

$$r_l = LayerNorm(x_l + ReLU(W_l \cdot x_l + b_l)), \quad l = 1, 2, 3 \tag{7}$$

$$\hat{y}_{price} = W_{out} \cdot r_3 + b_{out} \tag{8}$$

When the model is being trained on both yield and price at the same time, the overall loss function is a weighted sum of the yield MSE and a Huber loss on the price side. The weighting coefficient λ_p follows a warmup schedule:

$$L = L_{yield}^{MSE} + \lambda_p \cdot L_{price}^{Huber} \tag{9}$$

The Huber loss itself, with the threshold δ set to 0.5, is defined as:

$$L_{Huber} = (1/2)(y - \hat{y})^2 \qquad if\ |y - \hat{y}| \le \delta \tag{10}$$

$$L_{Huber} = \delta\,|y - \hat{y}| - (1/2)\,\delta^2 \qquad otherwise \tag{11}$$

During the backward pass, the gradients flowing into the price head are multiplied by a factor of 2.0 ($\nabla_{\theta p} L_{price} \leftarrow 2 \times \nabla_{\theta p} L_{price}$) to make sure the price branch gets enough of a training signal. For the first 50 epochs the model only learns from the yield loss ($\lambda_p = 0$), so it has time to get a reasonable handle on the supply side before the price

objective is switched on ($\lambda_p = 1$). In addition, an exponential moving average of the model weights with $\beta = 0.999$ is used, mixup augmentation with $\alpha = 0.2$ is applied, and a small amount of Gaussian noise is injected into the inputs. The hyperparameters that were settled on use 64 convolutional filters and an LSTM hidden size of 128 across 2 layers, feeding a fully connected hidden layer of 96 units, with dropout of 0.25, a learning rate of 0.001, and weight decay of 0.0003. For the final predictions a 5-seed ensemble is averaged.

6.7.2. Fusion Architectures

In Early Fusion, everything gets concatenated into one long feature vector before the model sees it:

$$x = [x_{satellite} \parallel x_{weather} \parallel x_{market} \parallel x_{lag} \parallel x_{static}] \quad (12)$$

Late Fusion works differently. It breaks the problem into three stages that run one after another:

$$\text{Stage 1: } \hat{y}_{yield} = f_{yield}(x_{crop}, x_{climate}) \quad (13)$$

$$\text{Stage 2: derive supply signals: predicted\_yield, yield\_anomaly, supply\_index, shortfall\_indicator} \quad (14)$$

$$\text{Stage 3: } \hat{y}_{price} = f_{price}(x_{market}, x_{supply\_signals}) \quad (15)$$

VII. DATASET INFORMATION

The framework needs four kinds of input, namely official yield statistics, wholesale market prices, satellite vegetation indices, and gridded weather. For the case study, each of these comes from a dataset that anyone can download for free. Table 2 lists what each one provides, at what spatial and temporal resolution, and how far back it goes.

Table 2. Public datasets used in the study

	Key Variables	Resolution	Coverage / Period
DES Agriculture Portal [19]	**Wheat yield (kg/ha), sown area, production**	**Annual, district level**	**2001–02 to 2024–25; 22 Haryana districts**
AGMARKNET via Kaggle [20]	**Mandi modal price (INR/quintal)**	**Weekly aggregate**	**Haryana FAQ-grade wheat; ~2001 to 2025**
Sentinel-2 + MODIS via GEE	**NDVI, EVI, NDWI**	**16-day composite; S2 from 2017 at 10–20 m**	**2001–2025 (MODIS 250 m); 2017–2025 (S2)**
ERA5-Land via GEE	**Temperature (2 m), relative humidity, wind speed**	**Hourly → weekly district mean**	**Global reanalysis, ~9 km grid; 2001–2025**

7.1. Yield Data: DES Agriculture Portal [19]

District-level wheat area, yield (kg/ha), and total production come from the Department of Economics and Statistics (DES) portal at data.desagri.gov.in [19]. The records span the 2001–02 through 2024–25 Rabi seasons across all 22 Haryana districts, that is, 528 district-year pairs. The state-level aggregate row that DES also publishes is deliberately omitted, since including it would allow the model to rely on a single Haryana-wide average rather than distinguishing districts.

7.2. Market Price / Mandi Price [20]

For daily wholesale prices (i.e., mandi prices), the Kaggle dataset Daily Commodity Prices – India is used, which mirrors AGMARKNET records back to roughly 2001. Each row includes the state, district, market, commodity, variety, grade, and date, along with that day's minimum, maximum, and modal prices in Indian Rupees per quintal. From these, only the wheat rows for Haryana at the FAQ grade are retained, and the modal prices are then averaged into a single weekly figure for each district-market pair.

7.3. Satellite Imagery: Sentinel-2 and MODIS

For the years from 2017 onward, we draw on Sentinel-2 Surface Reflectance Harmonised imagery (the Copernicus/S2_SR_HARMONIZED collection), which gives Level-2A data at 10 to 20-metre resolution. We mask clouds using the QA60 band and aggregate the remaining pixels from bands B2, B3, B4, B8, B11, and B12 to a single median value per district through Google Earth Engine's reduceRegion function. From these median reflectances we compute three vegetation indices, namely NDVI (the Normalised Difference Vegetation Index), EVI (the Enhanced Vegetation Index), and NDWI (the Normalised Difference Water Index). Because Sentinel-2 only goes back to 2017, a second satellite source is needed for the earlier years. For the period 2001 to 2025, MODIS MOD13Q1 version 6.1 is also used, which provides NDVI and EVI composites every 16 days at a 250-metre resolution. The SummaryQA layer is used to throw out low-quality pixels, and district-level averages are then computed in the same way. Both the Sentinel-2 and MODIS extractions are tied to FAO GAUL level-2 administrative boundaries so the district definitions stay consistent across the two sensors [21].

7.4. Weather Data: ERA5-Land

Weather data come from the ERA5-Land Hourly reanalysis product (ECMWF/ERA5_LAND/HOURLY), which has no gaps and covers the globe at a roughly 9 km grid spacing. Three variables are pulled: the temperature

at 2 metres above ground, relative humidity (derived from the dew point temperature), and wind speed at 10 metres. The hourly values are first aggregated to daily district-level means in Google Earth Engine, and then to weekly statistics. Each weekly record is tagged with a Rabi-season identifier spanning 1 October to 30 April, so that the weather data aligns with the annual yield figures published by DES [22].

VIII. EXPERIMENTAL RESULTS

8.1. Experimental Setup

Training is done for the years 2001 to 2023, yielding 484 district-year observations for the tree-based and CNN-LSTM models. TFT sees 461 of these, because the transformer needs a longer look-back window. Validation is done for the 2024 season, and testing is done for 2025, each covering all 22 districts. One caveat is worth keeping in mind throughout: because there are only 22 yield observations in the test set, an R^2 difference smaller than roughly 0.05 between two configurations carries little meaning. In total, five model families are constructed, namely XGBoost, Random Forest, a CNN-LSTM baseline, a modified CNN-LSTM with added multi-head attention, and TFT, and each one is run under both early and late fusion, resulting in 10 configurations [8]. The metrics reported on the test set are R^2, MAE, RMSE, and MAPE.

8.2. Test-Set Results

Tables 5a through 5d give the results on the 2025 test set. None of these 22 districts played any part in training or in model selection; they were held out from the beginning. On yield the strongest model turned out to be the Modified CNN-LSTM under Late Fusion, and on price the same network under Early Fusion. In each table the Modified CNN-LSTM row is highlighted for reference, and the two winning configurations are the ones just named.

Table 5a. Test-set yield results: Early Fusion (2025, n = 22)

Model	R^2	MAE (kg/ha)	RMSE (kg/ha)	MSE ($\times 10^3$)	MAPE (%)
XGBoost	0.7040	356.1	672.2	451.9	5.44
Random Forest	0.4856	424.2	886.1	785.2	6.29
CNN-LSTM Baseline	0.7523	274.5	614.8	378.0	4.01
Modified CNN-LSTM	0.7507	278.8	616.8	380.4	4.09
TFT Transformer	0.2966	413.9	1036.2	1073.7	5.76

Table 5b. Test-set yield results: Late Fusion (2025, n = 22)

Model	R^2	MAE (kg/ha)	RMSE (kg/ha)	MSE ($\times 10^3$)	MAPE (%)
XGBoost	0.7691	314.4	593.7	352.5	4.87
Random Forest	0.3585	505.5	989.5	979.1	7.58
CNN-LSTM Baseline	0.7619	277.8	602.8	363.4	4.13
Modified CNN-LSTM	0.7615	278.5	603.4	364.1	4.13
TFT Transformer	0.4401	408.5	924.5	854.7	5.90

Table 5c. Test-set price results: Early Fusion (2025, n = 22)

Model	R^2	MAE (INR/q)	RMSE (INR/q)	MSE ($\times 10^3$)	MAPE (%)
XGBoost	0.3342	247.6	368.9	136.1	8.22
Random Forest	0.5487	224.6	303.8	92.3	7.63
CNN-LSTM Baseline	0.7774	150.4	213.3	45.5	5.09
Modified CNN-LSTM	0.8207	129.5	191.5	36.7	4.34
TFT Transformer	0.3190	262.9	373.1	139.2	8.73

Table 5d. Test-set price results: Late Fusion (2025, n = 22)

Model	R^2	MAE (INR/q)	RMSE (INR/q)	MSE ($\times 10^3$)	MAPE (%)
XGBoost	0.0878	289.5	431.8	186.5	9.50
Random Forest	0.3426	336.3	366.6	134.4	12.24
CNN-LSTM Baseline	0.5699	272.7	296.5	87.9	9.95
Modified CNN-LSTM	0.6214	253.0	278.2	77.4	9.21
TFT Transformer	0.5907	165.8	289.3	83.7	5.15

The headline figures are as follows. The Modified CNN-LSTM under Late Fusion reaches $R^2 = 0.7615$, MAE = 278.5 kg/ha, RMSE = 603.4 kg/ha, and MAPE = 4.13% on yield; for price, under Early Fusion, the same network reaches $R^2 = 0.8207$ with MAE = 129.5 INR/quintal, RMSE = 191.5 INR/quintal, and MAPE = 4.34%. One detail needs mentioning here. Under Late Fusion, the CNN-LSTM Baseline actually posts a yield R^2 of 0.7619, against 0.7615 for the Modified variant, a gap of only 0.0004, which is far below the ±0.05 we treat as the threshold for a meaningful difference. The Modified variant is still the unified winner because it also provides

the best price forecast, and the simpler arrangement that covers both targets is the simpler arrangement. The Random Forest baseline sits well below both deep-learning winners, by about 0.27 in R^2 for price and around 0.40 for yield. XGBoost does post a higher test R^2 on yield (0.769), but this number cannot be taken at face value, because its training R^2 is essentially 1.000 and the model has simply memorised the training set. How little of that memorisation carries over to new data is visible under Late Fusion on price, where the same XGBoost goes from a training R^2 of 1.000 to a test R^2 of 0.088.

8.3. Comparison with Existing Literature

Table 6 places the present results next to figures reported in other published yield- and price-forecasting studies. Such comparisons need care. The region differs from study to study, and so do the crop calendar and the evaluation protocol, so the numbers are never strictly comparable. Keeping that in mind, the accuracy levels obtained in this work fall broadly within the same range as the strongest recent figures and, in several cases, exceed them.

Table 6. Comparison of test-set performance with published yield- and price-forecasting studies

Study	Target	Model	Region	R^2	RMSE	MAPE (%)
This work (Mod. CNN-LSTM, LF)	Yield	CNN-LSTM	Haryana, India	0.762	603 kg/ha	4.13
This work (Mod. CNN-LSTM, EF)	Price	CNN-LSTM	Haryana mandis	0.821	192 INR/q	4.34
Wolanin et al., 2020 [9]	Yield	1D-CNN	Indian Wheat Belt	≈0.80	—	≈7.0
Pandit et al., 2023 [7]	Yield	ARIMAX-LSTM	Rabi (India)	≈0.93	—	—
Sun et al., 2019 [11]	Yield	CNN-LSTM	US (soybean)	≈0.90	—	—
Maimaitijiang et al., 2020 [10]	Yield	UAV multimodal DL	US (soybean)	0.94	—	—
Manogna et al., 2025 [1]	Price	LSTM/GRU	India (multi)	—	−75% vs ARIMA	—
Nayak et al., 2024 [6]	Price	LSTM + exog. vars	India (multi)	—	−15–30% vs ARIMAX	—
Avinash et al., 2024 [5]	Price	HMM + DL ensemble	India (multi)	—	−20% in spikes	—
Nayak et al., 2025 [3]	Price	Meta-Transformer	India (multi)	—	—	9.0

On the yield side, the test R^2 of 0.762 sits just below the figure of roughly 0.80 that Wolanin et al. report for the Indian Wheat Belt [9], and below the approximately 0.93 reached by the ARIMAX-LSTM hybrid of Pandit et al. on Rabi crops [7]. Our MAPE of 4.13 per cent is lower than the 7 per cent MAE reported by Wolanin et al. [9] and falls in the same range as the best CNN-LSTM yield forecasts of Sun et al. on US soybean [11]. For price the comparison goes the other way. The test R^2 of 0.821 with MAPE of 4.34 per cent on Haryana mandi data is better than the Meta-Transformer result of Nayak et al. [3], who report a MAPE of around 9.0 per cent, and it holds up well next to the LSTM-with-exogenous-variables architecture of Nayak et al. [6], which reports 15 to 30 per cent RMSE reductions over ARIMAX but does not disclose an absolute R^2 value.

The more important observation, however, concerns the fact that every study in Table 6 addresses either yield or price in isolation, and none attempts both jointly. The framework described here addresses both through a single shared encoder, and the accuracy it reaches on each task is at least as high as that reported by these single-target systems. For this reason, combining the two forecasts within one pipeline does not appear to cost anything in accuracy.

8.4. Why Late Fusion Wins for Yield and Early Fusion Wins for Price

One observation that emerged clearly from the experiments is that yield and price do not favour the same fusion strategy. Late fusion with the modified CNN-LSTM produced the highest R^2 for yield (0.762); early fusion with the same architecture, on the other hand, was the most accurate configuration for price ($R^2 = 0.821$). Since this split in preference was observed consistently regardless of which model family was used, we consider it to reflect a genuine structural difference and not simply coincidence.

In the case of yield, late fusion outperforms early fusion. Our interpretation is that the two-stage design allows the yield model to first run on supply-side features and produce estimates of yield, surprise, and shortfall probability. These summary outputs then refine the final predictions without requiring the model to re-analyse raw data, and this also has the effect of making the results easier to interpret.

For price, the situation is reversed. Price prediction depends on fine-grained temporal details, for instance recent market shifts and volatility, and early fusion processes raw satellite and weather data directly, which allows it to capture these short-term events. Late fusion, by contrast, compresses the data into summary variables and in doing so loses some of the temporal patterns. Tree-based models like Random Forest can handle such summaries well enough, but the deep learning models clearly benefit when the raw temporal features are kept available.

In practical terms, what this means is that we end up running the same model twice. One run uses late fusion for yield and the other uses early fusion for price. All the data processing steps are shared between them; the only thing that changes is the way supply information enters the network. This arrangement lets each target receive whichever representation turns out to suit it better.

8.5. Diagnostic Diagrams

The figures below look at the results from two angles: how the ten model-fusion configurations compare on test R^2, and how closely the actual-versus-predicted scatter sits to the diagonal for the two winning models. Each figure is labelled with the specific model and target to which it refers.

The bar chart confirms what the tables have already established: the Modified CNN-LSTM under Late Fusion leads on yield with $R^2 = 0.762$, while the Modified CNN-LSTM under Early Fusion leads on price with $R^2 = 0.821$. The Random Forest baseline trails the deep-learning configurations on both targets, and XGBoost's strong nominal test figure on yield is an artefact of overfitting (training R^2 near 1.000).

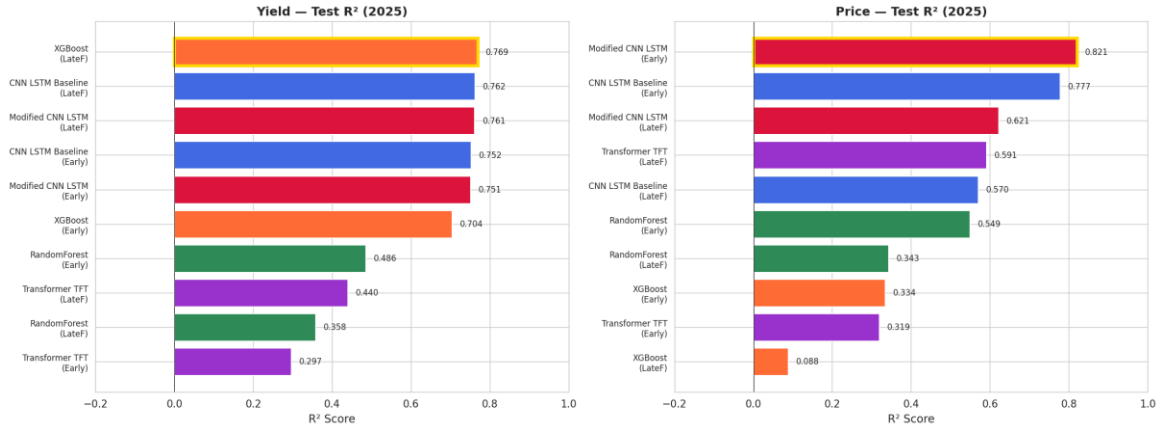

Fig. 2. Test-set R^2 comparison across all ten model-fusion configurations for yield (left) and price (right).

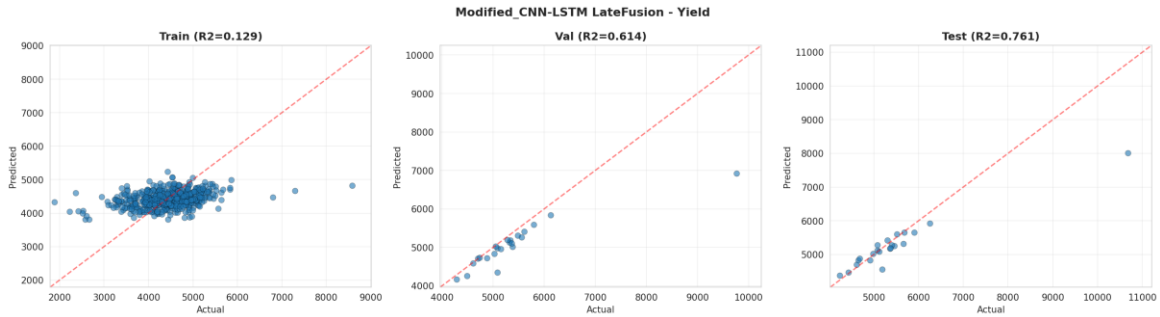

Fig. 3. Actual versus predicted: yield winner (Modified CNN-LSTM, Late Fusion). Dashed line = perfect prediction.

The training points (blue) are scattered around the diagonal with appreciable spread, as expected from stochastic mini-batch training in which each batch is a slightly different slice of the data. The validation points (orange) and test points (green) cluster considerably more tightly along the one-to-one line, which is an encouraging sign that the model generalises to data it was not trained on.

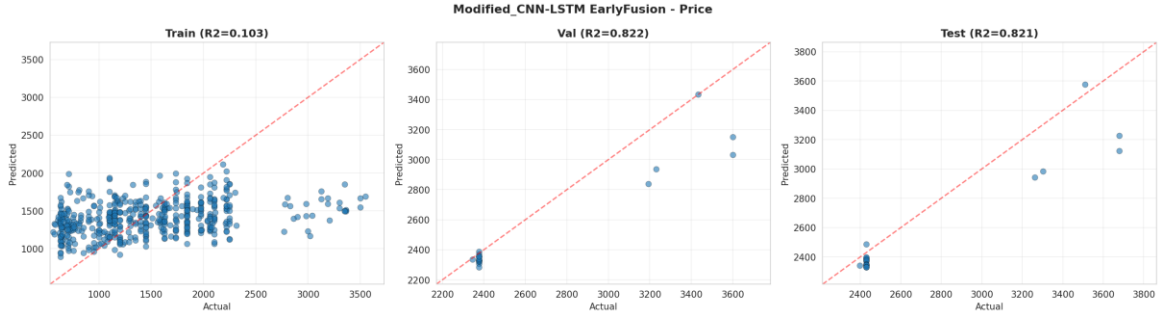

Fig. 4. Actual versus predicted: price winner (Modified CNN-LSTM, Early Fusion).

The price scatter is markedly tighter for this configuration than for any of the other price models. The few points lying above the diagonal in the high-price region correspond to districts where the observed price exceeded the model's expectation. These are most plausibly cases in which an MSP adjustment or local market conditions pushed the price beyond what the model's features could anticipate.

IX. DISCUSSION

A forecast is useful only when the people who depend on it can understand why the model predicted what it did. In the agricultural setting, a procurement officer who is sizing a buffer stock, or a policymaker who is weighing a revision of the MSP, will want to know that the model's reasoning rests on agronomically and economically sensible signals rather than spurious correlations. The Modified CNN-LSTM is the model deployed for both targets, so the examination here focuses on that model. For this purpose, two post hoc methods are used, each addressing a different question. Integrated gradients follow the contribution of every channel and timestep, upward or downward, so the source of a single forecast can be traced. LIME, on the other hand, reports what drove one particular prediction. The subsections below take these one at a time, for yield (Late Fusion) and for price (Early Fusion), because these are the two configurations in which the model will actually run.

9.1. Integrated Gradients (CNN-LSTM)

For a neural network the attribution has to be taken through the gradients, and the method used for doing this is Integrated Gradients, computed with the Captum library. The method starts from a blank all-zeros baseline and accumulates the gradient along the path up to the actual input which the model received; every prediction can then be traced back to the input features and the timesteps that produced it.

One channel dominates yield attribution: precipitation. precip_mm shows its strongest values during the mid-season composites C5 and C6, and relative humidity (rh_pct) keeps contributing positively from C6 through C9. This window covers January to March, the grain-filling period for wheat in Haryana, and moisture availability during grain filling is among the best-documented controls on final yield in this part of India. The satellite vegetation channels show smaller attributions over the same span, mostly positive. Temperature hardly appears at all in the test-year average. As far as this model is concerned, the season is mainly a question of moisture.

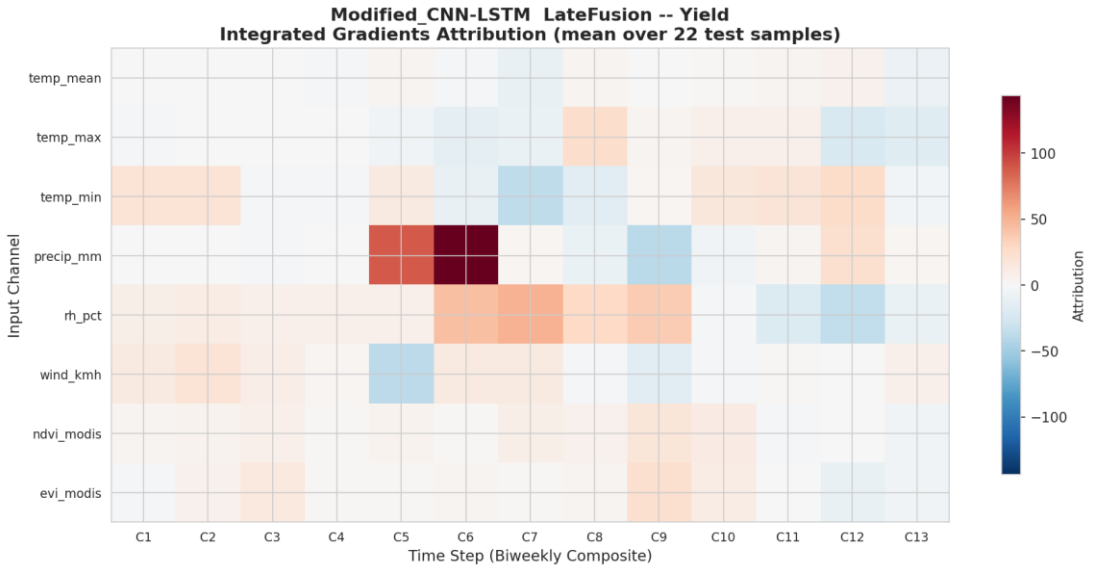

Fig. 5. Integrated gradients: yield (Modified CNN-LSTM, Late Fusion). The heatmap shows the mean attribution across 22 test samples, input channels, and timesteps.

9.2. Attribution for Price

Integrated gradients can be applied on the price configuration also, though one caveat has to be stated before that. The attribution here is computed only over the temporal weather and satellite input stack, which means it can tell how much those time-series channels move the price forecast and nothing beyond that. The market and policy scalars which the Early-Fusion price head also receives, namely the lagged and rolling mandi prices and the Minimum Support Price, stay outside the reach of this analysis.

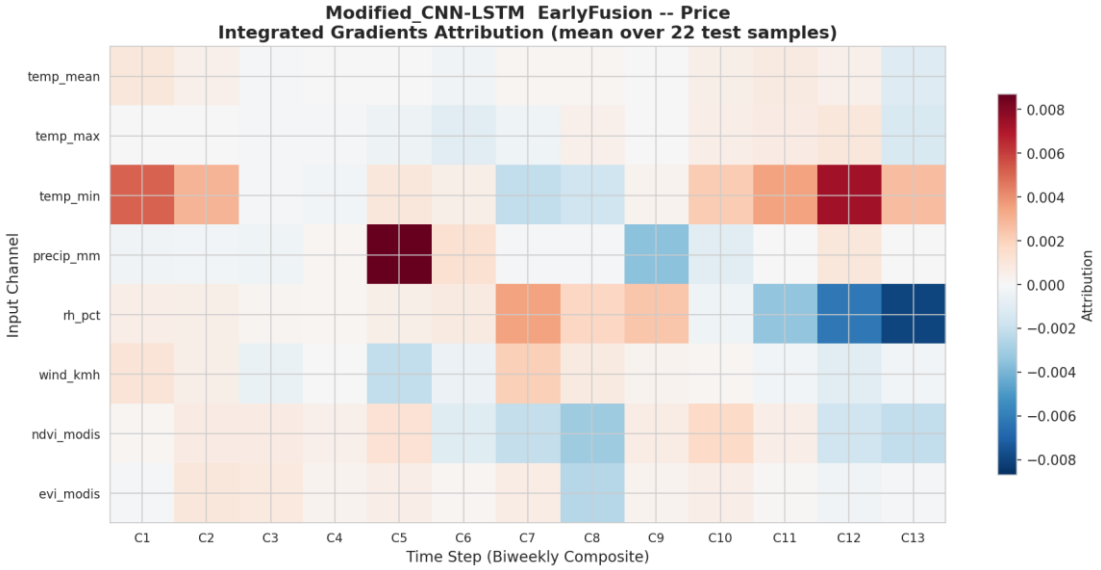

Fig. 6. Integrated gradients for the price model (Modified CNN-LSTM, Early Fusion), showing the attribution of the temporal weather and satellite channels averaged over the 22 test samples.

What comes back from this exercise is close to nothing, and that in itself says something. For every weather and satellite channel, and for every timestep, the attributions sit near zero, orders of magnitude below the corresponding values of yield, so the price forecast is essentially insensitive to the in-season weather and greenness streams. The signal of price is carried somewhere else, by the market and policy scalars (the recent mandi prices and the MSP) which the Early-Fusion head consumes along with the temporal stack; the temporal attribution simply cannot resolve them. So the useful content of this explanation is mostly of the negative kind; the model is not chasing any spurious weather correlation for price, and the behaviour of price follows the market history and the MSP anchor by construction.

9.3. LIME Local Explanations

Everything till now has been global, an average across predictions and timesteps. LIME works on one prediction at a time instead. The inputs of a single test case are nudged with small random changes, the shifts in the output are recorded, and a simple linear model is fitted on those shifts; the weights of this fitted model show which inputs will move that particular forecast the most. A representative yield case is shown in Figure 7.

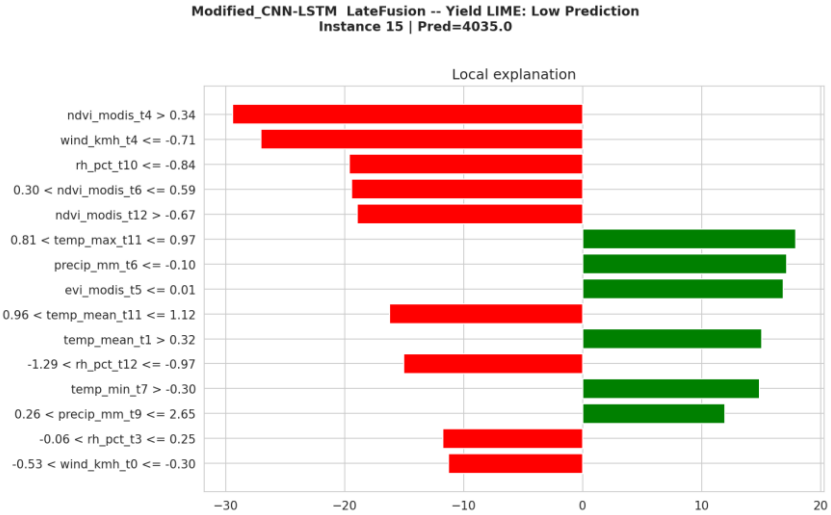

Fig. 7. LIME local explanation: yield prediction, low-yield district (Modified CNN-LSTM, Late Fusion). Green bars = features pushing prediction up; red bars = pushing down.

These instance-level explanations line up with the global integrated-gradients picture, which is reassuring. Take the low-yield district which was predicted at around 4,035 kg/ha. LIME attributes the outcome to a familiar mix of in-season signals; the mid-season NDVI and humidity readings on the unfavourable side pull

the prediction down, and the favourable windows of precipitation and temperature push it part of the way back up. Here also the forecast is driven by the weather and vegetation channels with no single dominant one among them, because the model integrates the whole season instead of hanging on one input. For the price configuration the caveat from above applies without any change. LIME perturbs only the temporal weather and satellite inputs, so the local explanations of price bring up those same channels with very small weights and not the market scalars which actually set the level of prices; the conclusion that price is governed by inputs outside the temporal stack only gets reinforced.

9.4. Summary of Explainability Findings

When the two methods are taken together, they describe the deployed Modified CNN-LSTM in a consistent way, target by target.

On yield, the two methods agree: the season is read mainly through mid-season moisture, that is, precipitation and humidity in the grain-filling window of January to March, with satellite vegetation channels in support. Decades of agronomic evidence on irrigated wheat in northern India say the same thing [4], [9], and a procurement officer gets a forecast that can be traced back to weather already being monitored. The story for price is useful in a different way. The temporal-channel attributions come out uniformly negligible, which means the model does not read price from the weather of the season at all; instead, the recent market history and the MSP anchor, which the Early-Fusion head consumes directly, govern the behaviour of price. This dependence follows from the construction of the model and from the behaviour of administered mandi prices [5], [6], even though the temporal attribution maps cannot resolve those scalar inputs. In practical terms, both of the targets end up at the same place: the forecasts rest on signals which a user already understands and trusts.

9.5. General Discussion

Combined, the performance of R^2 and MAPE measures derived from the forecasted values of yield and price proves to be as good as, if not better than, the numbers achieved by specialized systems that consider each target separately. An important practical benefit of our model is that both predictions come from the same pipeline and therefore no reconciliation is required. Late fusion effectively provides inductive prior of summary statistics to the yield head, which turns out to suit the yield variable perfectly. However, in case of price, the summary statistics tend to be limiting since prices contain information about detailed temporal patterns, which are lost with summarizing. An approach that was not considered and which might enhance results for the price forecast is to use raw tensors along with supply scalars. Naturally, there are certain drawbacks to our system. The DES yields dataset is aggregated to the level of districts, resulting in just 22 samples per year of testing. Moreover, lack of data in AGMARKNET mandi prices further adds noise to the process. Nevertheless, all four datasets used are available for public access and are refreshed regularly enough to enable weekly forecasts with one-week lag. Since the pipeline is parameterised by crop calendar, satellite collection identifiers, and mandi codes, it can be carried over to other crops and regions simply by adjusting these settings, while the core models and fusion logic remain unchanged.

X. CONCLUSIONS

The objective of this work was to develop a single, generalisable framework that forecasts crop yield and market price using only publicly available data, with predictions transparent enough for a non-technical user to understand the reasoning behind them. We validated the framework on wheat and mandi prices in Haryana, holding out the 2025 Rabi season across all 22 districts, and the results suggest that the objective is achievable. With late fusion, the modified CNN-LSTM reaches a test-set R^2 of 0.762 and a MAPE of 4.13 per cent for yield, and with early fusion the same architecture gives an R^2 of 0.821 and a MAPE of 4.34 per cent for price. Both sets of numbers are clearly above the Random Forest baseline. It is worth noting that XGBoost manages a comparable test R^2 on yield; however, it arrives at this by memorising the training set, as evidenced by a training R^2 of essentially 1.000, and on price it collapses entirely, making it a model that does not genuinely generalise. Across the price target and the harder fusion modes, the advantage of deep learning over tree-based baselines ranges from 0.20 to 0.70 R^2 points, confirming how critical it is to encode temporal patterns, a capability that tree-based models lack on their own [1], [4], [6], [7], [9], [18].

When it comes to explainability, the analysis of the deployed modified CNN-LSTM gives a consistent and interpretable picture. For yield, both integrated gradients and LIME are in agreement with each other: the forecast is driven mainly by mid-season moisture, above all precipitation and humidity during the grain-fill period, while the satellite vegetation indices play more of a supporting role. On the price side, the model pays no attention to in-season weather or satellite streams. Instead, the price forecast is governed by the recent history of market prices together with the MSP anchor that the price head consumes directly [5], [6].

From a design perspective, the architecture has been kept modular on purpose; nothing in it is hard-coded for wheat or for Haryana. Crop calendar, satellite collection identifiers, and AGMARKNET mandi codes are all

treated as parameters that a user can swap out, and once these are changed the rest of the system, namely the model structure, fusion logic, and explainability tools, carries over without the need for re-engineering. A natural direction for future work is to try the pipeline on other crops and regions across India, for instance rice in Punjab, sugarcane in Uttar Pradesh, and cotton in Gujarat, in order to see whether the accuracy obtained on Haryana wheat holds up in settings where growing conditions, market structure, and data quality are quite different. Additionally, it would be worth incorporating APMC procurement arrival volumes and commodity futures data, as these could help the price model in predicting both levels and direction more accurately. Looking further ahead, there is scope for extending the scenario engine to work with multi-year climate projections, which would allow users to examine how yield and price distributions might shift over the coming decade under different warming pathways.

AUTHOR CONTRIBUTIONS STATEMENT

Vivek Kumar conceived the original idea for the study, collected all datasets, wrote and ran the data-processing and model-training code, produced the figures and tables, and wrote the first draft of the paper.
Dr Rahul R. Chakre supervised the project throughout, provided guidance on the research direction at each stage, and reviewed and helped improve the manuscript.
Both authors have read and approved the final manuscript for publication.

CONFLICT OF INTEREST STATEMENT

The authors declare no competing interests.

FUNDING DECLARATION

No funding was received for this study.

DATA AVAILABILITY STATEMENT

All the data used in this study is freely available. The crop production numbers come from the Government of India's Directorate of Economics and Statistics, which publishes them at https://data.desagri.gov.in. The mandi prices were downloaded from a Kaggle dataset put together by Manas Khandelwal, available at https://www.kaggle.com/datasets/khandelwalmanas/daily-commodity-prices-india. For satellite imagery we used Google Earth Engine, and the weather variables were pulled from the ECMWF ERA5-Land hourly collection, also through Google Earth Engine.

DECLARATION OF GENERATIVE AI

Generative AI tools were used only to help with editing the language and phrasing of the manuscript, in line with IOSR Journals guidelines. All aspects of the research design, the data analysis, the interpretation of results, and the scientific conclusions are entirely the work of the authors.

ACKNOWLEDGMENTS

We want to thank the Directorate of Economics and Statistics under the Government of India for publishing their agricultural production data in a form that anyone can download and use. We are also grateful to ESA and NASA for making satellite imagery available through Google Earth Engine, and to ECMWF for the ERA5-Land reanalysis product, which we relied on heavily for the weather variables in this study.

ABBREVIATIONS

CNN — Convolutional Neural Network; DES — Directorate of Economics and Statistics; EF — Early Fusion; EVI — Enhanced Vegetation Index; GDD — Growing Degree Days; GEE — Google Earth Engine; LF — Late Fusion; LSTM — Long Short-Term Memory; MAE — Mean Absolute Error; MAPE — Mean Absolute Percentage Error; MODIS — Moderate Resolution Imaging Spectroradiometer; MSE — Mean Squared Error; MSP — Minimum Support Price; NDVI — Normalised Difference Vegetation Index; NDWI — Normalised Difference Water Index; QA — Quality Assurance; RF — Random Forest; RMSE — Root Mean Squared Error; SHAP — SHapley Additive exPlanations; TFT — Temporal Fusion Transformer; XAI — Explainable Artificial Intelligence; XGB — XGBoost.

REFERENCES

[1] V. S. Manogna, P. Raju, and K. Lavanya, "On the use of deep learning for improving agricultural commodity price forecasts," Scientific Reports, 2025.
[2] Y. Min, S. Kim, and J. Park, "Graph neural networks and recurrent architectures for international agricultural commodity price prediction," Scientific Reports, 2025.
[3] M. Nayak, B. Nayak, and P. K. Behera, "Agricultural commodity price forecasting via meta-transformer models tuned with meta-heuristic optimisation," Journal of Big Data, 2025.
[4] M. Ashfaq, H. Ali, and K. Rani, "Wheat yield prediction based on fusing climate variables with NDVI through machine learning," IEEE Access, vol. 12, 2024.

[5] G. Avinash, T. Ravi, and S. Banerjee, "Hidden Markov models as a guide for deep learning in the forecasting of highly volatile agricultural commodity prices," Applied Soft Computing, vol. 158, Art. no. 111557, 2024.
[6] M. Nayak, D. Mishra, and S. Panda, "Crop price forecasting in India: incorporating exogenous variables into deep learning models," Scientific Reports, 2024.
[7] A. Pandit, R. Singh, and M. Kaur, "A hybrid ARIMAX-LSTM framework and its application to Rabi crop yield modelling," Scientific Reports, 2023.
[8] A. Oikonomidis, C. Catal, and A. Kassahun, "Deep learning for crop yield prediction: a systematic literature review," New Zealand Journal of Crop and Horticultural Science, vol. 51, no. 1, pp. 1–26, 2023. doi: 10.1080/01140671.2022.2032213
[9] A. Wolanin et al., "Estimating and Understanding Crop Yields with Explainable Deep Learning in the Indian Wheat Belt," Environmental Research Letters, vol. 15, no. 2, Art. no. 024019, 2020.
[10] M. Maimaitijiang et al., "Soybean yield prediction from UAV using multimodal data fusion and deep learning," Remote Sensing of Environment, vol. 237, Art. no. 111599, 2020.
[11] J. Sun, L. Di, Z. Sun, Y. Shen, and Z. Lai, "County-Level Soybean Yield Prediction Using Deep CNN-LSTM Model," Sensors, vol. 19, no. 20, Art. no. 4363, 2019.
[12] P. Nevavuori, N. Narra, and T. Lipping, "Crop yield prediction with deep convolutional neural networks," Computers and Electronics in Agriculture, vol. 163, Art. no. 104859, 2019.
[13] H. Najjar, M. Miranda, M. Nuske, R. Roscher, and A. Dengel, "Explainability of subfield level crop yield prediction using remote sensing," IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, vol. 18, pp. 4141–4161, 2025. doi: 10.1109/JSTARS.2025.3528068
[14] U. Shafi et al., "Tackling food insecurity using remote sensing and machine learning-based crop yield prediction," IEEE Access, vol. 11, pp. 108640–108657, 2023. doi: 10.1109/ACCESS.2023.3321020
[15] S. K. Sharma et al., "Machine learning techniques for crop yield forecasting in the semi-arid (3A) zone of Rajasthan, India," Current Agriculture Research Journal, vol. 11, no. 3, 2024.
[16] K. P. S. Attwal, "Integrated machine learning model for wheat yield prediction using agronomic and meteorological factors: a case study from Punjab, India," International Journal of Agriculture and Food Science, vol. 7, no. 8, pp. 385–400, 2025. doi: 10.33545/2664844X.2025.v7.i8f.636
[17] J. You, X. Li, M. Low, D. Lobell, and S. Ermon, "Deep Gaussian Process for Crop Yield Prediction Based on Remote Sensing Data," in Proc. 31st AAAI Conf. on Artificial Intelligence (AAAI), 2017, pp. 4559–4565.
[18] K. Kuwata and R. Shibasaki, "Estimating crop yields with deep learning and remotely sensed data," in Proc. IEEE Int. Geoscience and Remote Sensing Symp. (IGARSS), 2015, pp. 858–861.
[19] Government of India, Directorate of Economics and Statistics (DES), Agriculture Data Portal, 2025. [Online]. Available: https://data.desagri.gov.in
[20] M. Khandelwal, "Daily Commodity Prices – India," Kaggle Dataset, 2024. [Online]. Available: https://www.kaggle.com/datasets/khandelwalmanas/daily-commodity-prices-india
[21] J. S. Bojanowski et al., "Integration of Sentinel-3 and MODIS vegetation indices with ERA-5 agro-meteorological indicators for operational crop yield forecasting," Remote Sensing, vol. 14, no. 5, Art. no. 1238, 2022. doi: 10.3390/rs14051238
[22] K. Murakami et al., "Machine learning for municipality-level winter wheat yield prediction from meteorological data," PLoS ONE, vol. 16, no. 10, e0258677, 2021.
[23] S. Bouskour et al., "Wheat yield patterns and predictions in Taounate, Morocco, using MODIS-NDVI," IJEECS, vol. 37, no. 1, pp. 648–659, 2025.