A Deep Neural Network Approach For The Enhancement Of Prediction Capabilities Of Banana Harvesting Products

Agricultural yield prediction plays a critical role in optimizing resource allocation and guiding market planning. In this research, we focus on improving the accuracy of banana production forecasts for Colombia, where banana farming is a significant economic activity. We explore the application of cutting-edge machine learning techniques, including Gated Recurrent Units (GRU), Hybrid models, Bidirectional Long Short-Term Memory (Bi-LSTM), and the Prophet forecasting tool.

Using data collected from 10 banana plantations, we applied these models to predict production outcomes, aiming to surpass the accuracy of traditional time-series models such as ARIMA and SARIMA. Among the models, our Hybrid approach—combining RNN, Random Forest, and XGBoost—demonstrated the best performance, significantly improving key metrics like RMSE, MAE, and R2 scores. Prophet was employed to provide long-term forecasting, yielding valuable insights for future banana production trends.

Our findings indicate that these advanced machine learning models, particularly hybrid approaches, can greatly enhance the predictive capabilities for banana production. This holds promising implications for decision-making in the agricultural sector, facilitating more efficient resource planning, especially for export and domestic consumption markets.

***Vedika Shrikant Nalawade** and Carlos D. Paternina-Arboleda*

This research is supported by the Computational Science Research Center (CSRC) at San Diego State University

Gated Recurrent Unit (GRU)

GRU is a type of RNN designed to handle sequential data by efficiently capturing long-term dependencies without the vanishing gradient problem. It uses update and reset gates to control the flow of information, making it faster and simpler than traditional LSTMs while maintaining performance.

Model Components:

- **Update Gate:** Controls how much past information to retain.

$$z_t = \sigma(W_z \cdot [h_{t-1}, x_t])$$

- **Reset Gate:** Determines how much past information to forget.

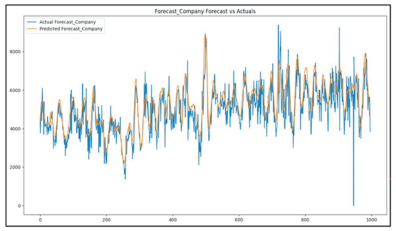
$$r_t = \sigma(W_r \cdot [h_{t-1}, x_t])$$

- **Candidate Hidden State:** Computes new memory content.

$$\tilde{h}_t = \tanh(W_h \cdot [r_t * h_{t-1}, x_t])$$

- **Final Hidden State:** Combines past and present information.

$$h_t = (1 - z_t) * h_{t-1} + z_t * \tilde{h}_t$$



The GRU model demonstrated robust performance in predicting Forecast_Company, effectively tracking real trends.

- **RMSE:** 502.03
- **R^2 Score**: 0.82
- **Result:** The model captured complex temporal dependencies, showcasing its ability to handle non-linear patterns in agricultural forecasting and providing reliable predictions for production planning.

Bidirectional Long Short-Term Memory (LSTM)

Bidirectional LSTM processes data in both forward and backward directions, capturing dependencies from past and future time steps. This improves its ability to learn complex patterns in sequential data, enhancing prediction accuracy.

Model Components:

- **Forward LSTM:** Processes the sequence from the start to the end.
- **Backward LSTM:** Processes the sequence from the end to the start, providing additional context.

Formulas:

- **Forward LSTM:**

$$\overrightarrow{h_t} = LSTM(x_t, \overrightarrow{h_{t-1}})$$

- **Backward LSTM:**

$$\overleftarrow{h_t} = LSTM(x_t, \overleftarrow{h_{t+1}})$$

- **Final Output:**

$$h_t = [\overrightarrow{h_t}, \overleftarrow{h_t}]$$

	Real_Packing				Prod_real_Ha				Forecast_Company			
Metric	GRU	Hybrid	Prophet	Bi-LSTM	GRU	Hybrid	Prophet	Bi-LSTM	GRU	Hybrid	Prophet	Bi-LSTM
RMSE	527.82	537.33	409.53	537.33	6.89	6.69	6.69	6.89	502.03	604.47	719.83	719.83
MAE	395.76	407.44	302.06	597.5	5.41	5.14	5.31	5.41	307.16	432.45	543.8	543.8
Explained Variance Score	0.77	0.8	0.86	0.99	0.53	0.55	0.6	0.99	0.82	0.74	0.7	0.99
Mean Squared Log Error	0.01	0.01	0.01	0.01	0.02	0.02	0.02	0.02	0.08	0.09	0.1	0.01
R2 Score	0.76	0.75	0.86	-0.55	0.52	0.55	0.54	-0.91	0.82	0.74	0.63	-0.18

The Bidirectional LSTM model struggled with predicting Real_Packing and Forecast_Company, with negative R^2 scores (-0.55 and -0.18) and higher RMSE values, indicating potential overfitting and data-specific issues.

- **Result:** Despite these challenges, the model still highlights the potential of capturing both past and future dependencies, but requires further tuning to improve accuracy.
- The comparison table shows the model's performance metrics, underlining areas where improvement is needed for better prediction accuracy.

PROPHET

Prophet is a time series forecasting model developed by Facebook, designed to handle complex data with strong seasonal trends. It uses an additive model that incorporates trend, seasonality, and holiday effects, making it robust against missing data and capable of providing uncertainty intervals to improve decision-making.

Model Components:

- **Additive Model:** The forecast is the sum of trend, seasonality, and holiday effects:

$$y(t) = g(t) + s(t) + h(t) + \epsilon_t$$

where:

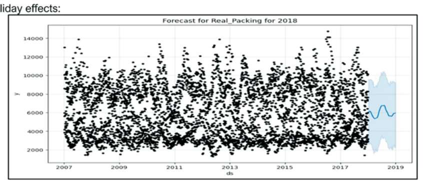
- **g(t):** Trend function capturing non-periodic changes.
- **s(t):** Seasonal component for periodic effects (e.g., yearly or weekly).
- **h(t):** Holiday effects (optional).
- **ϵ(t):** Error term representing noise.
- **Trend Function:**

$$g(t) = \alpha + \beta t$$

where **α** and **β** are trend parameters.

- **Seasonality Function:**

$$s(t) = A\sin(2\pi t/P) + B\cos(2\pi t/P)$$



where **P** is the period, and **A** and **B** are seasonality coefficients.

- **Uncertainty Intervals:** Prophet provides uncertainty intervals for the forecast, based on future trend and seasonality predictions.

Model Results :

- The **Prophet model** performed exceptionally well in forecasting banana production:
- **RMSE:** 409.53
- **MAE:** 302.06
- **R^2 Score:** 0.86

Result:

- The model outperformed other techniques, especially for **Real_Packing**, by effectively capturing trends and seasonality.
- The plot of **Prophet Model: Predicted vs Actual values for Real_Packing** shows strong alignment, indicating the model's ability to capture critical patterns in the time series.

HYBRID MODEL - RNN,Random Forest & XGBoost

Recurrent Neural Network (RNN):

RNN captures temporal dependencies in sequential data like time series, making it ideal for agricultural yield prediction.

Process:

- The input at time step t is processed with the hidden state from the previous time step to update the current hidden state.
- **Formula:** $h_t = \tanh(W_{hh}h_{t-1} + W_{xh}x_t + b_h)$
- **Output:** The hidden state ht is used to predict the banana yield at each time step.

Random Forest:

Random Forest is an ensemble method that reduces overfitting and handles high-dimensional data well.

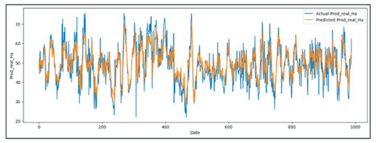
Process:

- Multiple decision trees are trained on random subsets of data.
- The final prediction is the average of the trees outputs.

Formula:

$$H(x) = \frac{1}{T}\sum_{i=1}^{T} h_i(x)$$

- **Output:** Random Forest predicts residual errors



XGBoost:

XGBoost is a boosting algorithm that corrects prediction errors sequentially.

Process:

- Each new tree in the sequence is trained to correct the residuals of the previous trees.
- **Formula:** $\hat{y}^{(m)} = \hat{y}^{(m-1)} + \alpha f_m(x)$
- **Output:** XGBoost further refines the predictions by reducing residuals from Random Forest.

- The hybrid model combining RNN, Random Forest, and XGBoost effectively predicted Prod_real_Ha with:
- **RMSE:** 6.69
- **MAE:** 5.31
- **R^2 Score:** 0.54
- It captures temporal dependencies, reduces overfitting, and refines residuals, significantly improving prediction accuracy over traditional methods.