Parag Yadav, 2026, 14:3
ISSN (Online): 2348-4098
ISSN (Print): 2395-4752

International Journal of Science, Engineering and Technology
An Open Access Journal

An End-to-End Deep Learning Framework for Sugarcane Quality Assessment Based on Near-Infrared Spectroscopy and Explicit Feature Interaction-Aware Graph Neural Networks

Parag Yadav, Assistant Professor Raj Kumar, Shantanu Yadav, Tanya Rana, Yash Singh Kathayat
Department of Computer Science and Engineering, Quantum University, Roorkee, India

Abstract- Accurate, rapid, and non-destructive assessment of sugarcane quality is a critical requirement for optimizing harvesting schedules, refinery throughput, and fair payment systems in the global sugar industry. Conventional wet chemistry methods, while accurate, are time-consuming, reagent-intensive, and unsuitable for real-time field deployment. Near-Infrared (NIR) spectroscopy has emerged as a powerful analytical technique for non-destructive quality measurement; however, existing machine learning approaches applied to NIR spectral data predominantly rely on shallow models or architectures that treat samples in isolation, thereby neglecting the rich relational and interaction structure latent within spectral populations. In this paper, we propose a novel end-to-end deep learning framework, designated EFI-GNN-NIR, that integrates 1D Convolutional Neural Network (CNN)-based spectral feature extraction with an Explicit Feature Interaction-Aware Graph Neural Network (EFI-GNN) for simultaneous prediction of Brix percentage, Pol percentage, and purity index of sugarcane samples. The proposed framework constructs a population-level spectral similarity graph wherein each node represents an individual NIR sample and edges encode cosine similarity relationships between learned spectral embeddings. An explicit feature interaction module, employing bilinear cross-network operations within the graph message-passing paradigm, enables the model to capture higher-order cross-wavelength dependencies that conventional models overlook. The framework is evaluated on a benchmark NIR sugarcane dataset augmented with agronomic metadata, demonstrating superior performance with R2=0.978R^2 = 0.978 R2=0.978, RMSE=0.142RMSE = 0.142 RMSE=0.142 Brix, and RPD=6.84RPD = 6.84 RPD=6.84 for Brix prediction, outperforming state-of-the-art baselines including Partial Least Squares Regression (PLS-R), Support Vector Regression (SVR), Random Forest (RF), standard CNN, and vanilla GNN architectures. Multi-task learning across three quality parameters yields consistent improvements over single-task counterparts. This work provides a significant contribution toward precision agriculture, smart sugarcane farming, and real-time quality monitoring pipelines deployable at factory intake points.

Keywords— Near-Infrared Spectroscopy; Sugarcane Quality Assessment; Graph Neural Networks; Explicit Feature Interaction; Deep Learning; Brix Prediction; Precision Agriculture; Spectral Feature Extraction; Multi-Task Learning; Smart Farming

I. INTRODUCTION

1. Background and Motivation

Sugarcane (Saccharum officinarum) remains one of the world's most economically significant crops, accounting for approximately 80% of global sugar production and serving as a primary feedstock for bioethanol, bioplastics, and fiber industries [1]. Global sugarcane production exceeded 1.9 billion metric tonnes in 2022, with major producing nations including Brazil, India, China, Thailand, and Australia contributing the majority of output [2]. The economic value of harvested sugarcane is directly determined by its sucrose content and associated

quality parameters, principally the Brix value (total dissolved solids as a percentage), Pol percentage (apparent sucrose content), and purity index (ratio of Pol to Brix). Accurate determination of these parameters at harvest and at mill intake is fundamental to fair payment systems, blending decisions, and process optimization [3].

Conventional analytical methods for sugarcane quality assessment, including wet chemistry procedures such as polarimetry, refractometry, and high-performance liquid chromatography (HPLC), are well-established and accurate. However, these methods are inherently destructive, require laboratory infrastructure, involve chemical reagents, and typically demand 30–90 minutes per sample — rendering them impractical for high-throughput field or mill-side deployment [4]. As global agricultural supply chains increasingly demand real-time quality data to support precision farming decisions, there exists a pressing need for non-destructive, rapid, and automated quality assessment methodologies.

2. Near-Infrared Spectroscopy in Agriculture

Near-Infrared (NIR) spectroscopy operates in the electromagnetic wavelength range of approximately 800–2500 nm, where overtones and combination bands of molecular vibrations — particularly those associated with C-H, O-H, and N-H bonds — produce characteristic absorption signatures [5]. Because sucrose, water, cellulose, and other sugarcane constituents exhibit distinct NIR absorption patterns, NIR spectroscopy has been extensively studied as a rapid, non-destructive alternative to wet chemistry for quality assessment [6]. NIR instruments can acquire full spectral profiles within seconds, require minimal sample preparation, and can be readily integrated into conveyor-belt or flow-cell configurations for real-time inline monitoring [7].

Despite these advantages, the primary challenge in NIR spectroscopy lies in the extraction of meaningful quantitative predictions from high-dimensional, collinear spectral data. A typical NIR spectrum spanning 800–2500 nm at 2 nm resolution yields 851 data points per sample, with substantial multicollinearity across adjacent wavelengths. Classical chemometric approaches, particularly Partial Least Squares Regression (PLS-R), have dominated NIR data analysis for decades and continue to serve as industry benchmarks [8]. However, PLS-R relies on linear latent variable decomposition and may fail to capture the complex nonlinear relationships and higher-order interactions present in biological spectral matrices.

3. Deep Learning and Graph Neural Networks

The advent of deep learning has substantially expanded the analytical capacity applicable to spectral data. Convolutional Neural Networks (CNNs) applied along the spectral dimension have demonstrated strong performance in capturing local spectral patterns, while recurrent architectures have been applied to model sequential spectral dependencies [9]. More recently, Transformer-based models with self-attention mechanisms have been adapted for spectral regression tasks, achieving state-of-the-art results on several agricultural quality benchmarks [10].

Graph Neural Networks (GNNs), a class of deep learning architectures operating directly on graph-structured data, have emerged as powerful tools for learning over relational datasets [11]. In agricultural and biological domains, GNNs have been applied to molecular property prediction, protein structure analysis, and crop disease detection [12]. A critical insight motivating the present work is that a population of NIR spectral samples is not a collection of independent vectors — rather, samples from similar varieties, growing conditions, or maturity stages share spectral affinities that constitute meaningful relational structure. Representing this structure as a graph and performing inference over the graph enables collective learning that isolated models cannot achieve.

4. Research Objectives

This paper addresses the following objectives:

- Develop an end-to-end deep learning framework that unifies spectral preprocessing, 1D-CNN feature extraction, spectral similarity graph construction, and GNN-based regression for sugarcane quality assessment.

- Design an Explicit Feature Interaction (EFI) module within the GNN architecture that captures higher-order cross-spectral feature interactions through bilinear cross-network operations.
- Implement multi-task learning across Brix, Pol, and purity prediction within a unified model architecture.
- Conduct comprehensive benchmarking against established chemometric and machine learning baselines, demonstrating the superiority of the proposed approach.

5. Paper Organization

The remainder of this paper is organized as follows: Section 2 reviews related literature; Section 3 formally states the problem; Sections 4–10 detail the proposed methodology; Sections 11–13 describe experiments and results; Sections 14–16 discuss advantages, limitations, and future work; Section 17 concludes the paper.

II. LITERATURE REVIEW

1. NIR Spectroscopy for Sugarcane Quality Assessment

The application of NIR spectroscopy to sugarcane quality measurement has a research history spanning over three decades. Early work by Berding and Brotherton [13] established the feasibility of NIR-based Brix prediction using near-infrared diffuse reflectance. Subsequent studies by Tillmann et al. demonstrated that PLS-R applied to NIR spectra of sugarcane juice could achieve prediction errors within acceptable industrial tolerances, with RMSEP values below 0.25 Brix for juice samples [14].

More recent investigations have extended NIR analysis to intact stalks, bagasse, and powdered samples. Liu et al. [15] reported R2=0.93R^2 = 0.93 R2=0.93 for Brix prediction from intact sugarcane NIR spectra using PLS-R with multiplicative scatter correction (MSC) preprocessing. Guo et al. [16] applied support vector regression (SVR) to preprocessed NIR spectra and demonstrated improvements over PLS-R baselines in Pol percentage prediction, particularly for heterogeneous sample sets. The consensus from this body of literature is that NIR spectroscopy is analytically viable for sugarcane quality prediction, but predictive performance depends critically on preprocessing protocols and the capacity of the modeling approach to handle spectral nonlinearity and collinearity.

2. Machine Learning Approaches in Spectral Analysis

Classical machine learning methods applied to NIR spectral data include Principal Component Regression (PCR), PLS-R, SVR, and ensemble methods such as Random Forest (RF) and Gradient Boosting. Among these, PLS-R remains the most widely adopted, primarily because it efficiently handles the ill-posed regression problem posed by high-dimensional collinear spectral data [8]. However, PLS-R is fundamentally a linear method and cannot model complex nonlinear spectral-quality relationships without explicit kernel transformations.

Random Forest and gradient boosting approaches have shown improved nonlinear modeling capacity but suffer from the curse of dimensionality when applied directly to raw high-dimensional spectra without dimensionality reduction [17]. Feature selection methods, including Variable Importance in Projection (VIP) from PLS-R and genetic algorithm-based wavelength selection, have been employed to reduce spectral dimensionality prior to ensemble modeling, yielding competitive results in some studies [18].

3. Deep Learning for Agricultural Quality Assessment

The application of deep learning to agricultural quality assessment has accelerated substantially since 2019. Zhang et al. [19] proposed a 1D-CNN architecture for near-infrared spectral regression, demonstrating that convolutional filters could learn spectral feature detectors without manual wavelength selection. Their model achieved R2=0.96R^2 = 0.96 R2=0.96 for wheat protein content prediction. Cui and Fearn [20] conducted a systematic comparison of deep learning methods for NIR spectroscopy, finding that CNNs outperformed

PLS-R on complex, multi-constituent datasets while performing comparably on simpler regression tasks. For sugarcane specifically, Nawi et al. [21] applied artificial neural networks (ANNs) to NIR spectral data for sucrose content prediction, reporting R2=0.94R^2 = 0.94 R2=0.94 with a three-layer feedforward architecture. More recently, Transformer-based spectral models have been explored for food quality assessment, with Dosovitskiy-inspired spectral transformers achieving strong results on fruit quality datasets [22]. However, these architectures treat each sample independently and do not exploit relational structure across sample populations.

4. Graph Neural Networks in Scientific Domains

Graph Neural Networks were formalized by Kipf and Welling [23] through the Graph Convolutional Network (GCN) framework, subsequently extended by Veličković et al. [24] via Graph Attention Networks (GATs) that incorporate learnable attention weights on edges. The GraphSAGE framework by Hamilton et al. [25] introduced inductive learning on graphs, enabling generalization to unseen nodes — a critical capability for deployment on new sample batches.

In the agricultural and food science domain, GNNs have been applied to plant phenotype prediction [26], soil property mapping [27], and food adulteration detection [28]. Notably, GNN applications in spectroscopy remain scarce. Chen et al. [29] constructed molecular graphs from chemical structures and predicted NIR absorption features, but the inverse problem — constructing sample-level graphs from NIR data for quality prediction — has not been systematically explored.

5. Feature Interaction in Deep Learning

Explicit feature interaction modeling has been extensively studied in recommendation systems and click-through rate prediction, with models such as DeepFM [30], CrossNet [31], and HOFM demonstrating that explicitly modeling pairwise and higher-order feature interactions substantially improves predictive performance over implicit interaction models. The integration of cross-network mechanisms into GNN architectures represents a relatively unexplored research direction. The proposed EFI-GNN adapts these principles to the spectral domain, incorporating explicit bilinear feature interactions within the graph message-passing framework — to the best of the authors' knowledge, this is the first such application in NIR spectroscopy for agricultural quality assessment.

III. PROBLEM STATEMENT

1. Formal Problem Definition

Let D={(xi,yi)}i=1N\mathcal{D} = \{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^{N} D={(xi,yi)}i=1N denote a dataset of NN N sugarcane NIR spectral samples, where xi∈RL\mathbf{x}_i \in \mathbb{R}^{L} xi∈RL is the raw reflectance spectrum with LL L wavelength channels (e.g., L=851L = 851 L=851 for the range 800–2500 nm at 2 nm resolution), and yi=[yiBrix,yiPol,yiPur]∈R3\mathbf{y}_i = [y_i^{\text{Brix}}, y_i^{\text{Pol}}, y_i^{\text{Pur}}] \in \mathbb{R}^3 yi=[yiBrix,yiPol,yiPur]∈R3 is the corresponding multi-target quality vector. Additionally, each sample may carry agronomic metadata mi∈RM\mathbf{m}_i \in \mathbb{R}^{M} mi∈RM (e.g., variety code, harvest age, GPS coordinates).

The objective is to learn a function:

fθ:RL×RM→R3f_\theta: \mathbb{R}^L \times \mathbb{R}^M \rightarrow \mathbb{R}^3fθ:RL×RM→R3

parameterized by θ\theta θ, that minimizes the multi-task regression loss:

L=∑t∈{B,P,U}λt·1N∑i=1N(yit−y^it)2+α‖θ‖22\mathcal{L} = \sum_{t \in \{B, P, U\}} \lambda_t \cdot \frac{1}{N} \sum_{i=1}^{N} (y_i^t - \hat{y}_i^t)^2 + \alpha \|\theta\|_2^2L=t∈{B,P,U}∑λt·N1i=1∑N(yit−y^it)2+α‖θ‖22

where λt\lambda_t λt are task-specific loss weighting coefficients and α\alpha α is the L2 regularization coefficient.

The proposed framework additionally models this as a graph-level regression problem. Specifically, given a constructed graph G=(V,E)\mathcal{G} = (\mathcal{V}, \mathcal{E}) G=(V,E) where nodes V={vi}i=1N\mathcal{V} = \{v_i\}_{i=1}^N V={vi}i=1N

represent samples and edges E\mathcal{E} E encode spectral similarity, the GNN learns:
fθ:(G,X,M)→Y^f_\theta: (\mathcal{G}, \mathbf{X}, \mathbf{M}) \rightarrow \hat{\mathbf{Y}}fθ:(G,X,M)→Y^
where X∈RN×d\mathbf{X} \in \mathbb{R}^{N \times d} X∈RN×d is the matrix of spectral embeddings and M∈RN×M\mathbf{M} \in \mathbb{R}^{N \times M} M∈RN×M is the agronomic metadata matrix.

2. Identified Research Gaps

Gap ID	Description
G1	Existing NIR-based sugarcane models do not exploit inter-sample relational structure
G2	No end-to-end trainable framework integrating spectral encoding and graph-based inference exists for sugarcane
G3	Explicit feature interaction in spectral data remains unexplored in GNN contexts
G4	Multi-task prediction of Brix, Pol, and purity is rare in the literature
G5	Spectral graph construction strategies are ad hoc and not principled

IV. PROPOSED METHODOLOGY

1. Framework Overview

The EFI-GNN-NIR framework consists of five interconnected modules operating in a fully differentiable, end-to-end trainable pipeline:

- Spectral Preprocessing Module (SPM): Applies signal correction and normalization to raw NIR spectra.
- Spectral Feature Extractor (SFE): A 1D-CNN encoder that maps preprocessed spectra to compact latent embeddings.
- Graph Construction Module (GCM): Constructs a k-nearest neighbor (kNN) spectral similarity graph over the embedding space.
- Explicit Feature Interaction-Aware GNN (EFI-GNN): Performs graph-based inference with explicit cross-feature interaction layers.
- Multi-Task Prediction Head (MTPH): Produces simultaneous predictions of Brix, Pol, and purity via task-specific output branches.

2. End-to-End Pipeline

The complete data flow is:
xi→SPMx~i→SFEhi(0)→GCMG→EFI-GNNhi(K)→MTPHy^i\mathbf{x}_i \xrightarrow{\text{SPM}} \tilde{\mathbf{x}}_i \xrightarrow{\text{SFE}} \mathbf{h}_i^{(0)} \xrightarrow{\text{GCM}} \mathcal{G} \xrightarrow{\text{EFI-GNN}} \mathbf{h}_i^{(K)} \xrightarrow{\text{MTPH}} \hat{\mathbf{y}}_ixiSPMx~iSFEhi(0)GCMGEFI-GNNhi(K)MTPHy^i

All modules are jointly optimized during training via backpropagation through the multi-task loss.
Pseudo-code: EFI-GNN-NIR Training Loop
Algorithm 1: EFI-GNN-NIR End-to-End Training

Input: Dataset D = {(x_i, m_i, y_i)}, hyperparameters (k, d, K, λ, α, lr)
Output: Optimized model parameters θ*

1: Initialize θ randomly (He initialization for CNNs, Xavier for GNN)
2: FOR each epoch e = 1, ..., E DO
3: FOR each mini-batch B ⊂ D DO
4: /* Preprocessing */
5: x̃_i ← SPM(x_i) ∀i ∈ B // MSC + SNV + SG + Norm
6: /* Spectral Feature Extraction */
7: h_i^(0) ← SFE(x̃_i; θ_SFE) ∀i ∈ B // 1D-CNN encoder
8: /* Concatenate with metadata */
9: h_i^(0) ← [h_i^(0) ‖ m_i] ∀i ∈ B
10: /* Graph Construction */
11: G ← kNN_Graph({h_i^(0)}, k) // Cosine similarity
12: /* EFI-GNN Forward Pass */
13: FOR layer l = 0, ..., K-1 DO
14: /* Graph Attention */
15: h̃_i^(l) ← GAT_Layer(h_i^(l), N(i), G)
16: /* Explicit Feature Interaction */

```
    h_i^(l+1) ← EFI_Layer(h̃_i^(l), h_i^(0))
    /* Residual connection */
    h_i^(l+1) ← h_i^(l+1) + h_i^(l)  (if dims match)
  END FOR
  /* Graph Readout */
  z ← GlobalMeanPool({h_i^(K)})
  /* Multi-Task Prediction */
  ŷ^Brix, ŷ^Pol, ŷ^Pur ← MTPH(z; θ_MTPH)
  /* Loss Computation */
  L ← Σ_t λ_t · MSE(y^t, ŷ^t) + α||θ||²
  /* Backpropagation */
  θ ← AdamW_Update(θ, ∇_θ L, lr)
 END FOR
 lr ← CosineAnneal(lr, e, E)
END FOR
RETURN θ* = argmin_θ L
```

V. DATASET DESCRIPTION

1. Primary Dataset

The primary experimental dataset comprises NIR reflectance spectra collected from sugarcane juice and stalk samples across two crop seasons. Samples were collected from multiple varieties and harvesting ages to ensure diversity. The dataset characteristics are summarized in Table 1.

Table 1: Dataset Summary

Parameter	Value
Total Samples	1,240
Spectral Range	800–2500 nm
Wavelength Resolution	2 nm
Spectral Channels (L)	851
Brix Range	14.2 – 23.8 (%)
Pol Range	10.1 – 20.9 (%)
Purity Range	68.4 – 91.3 (%)
Varieties Represented	8
Sample Types	Juice (n=820), Intact Stalk (n=420)
Train / Val / Test Split	70% / 15% / 15%
Reference Method	Polarimetry + Refractometry
Instrument	Bruker MPA II FT-NIR

2. Agronomic Metadata Features

In addition to spectral data, each sample carries M=6M = 6 M=6 agronomic metadata features: variety code (one-hot encoded), harvest age in weeks, field pH, irrigation type (binary), geographic zone (encoded), and sample temperature at acquisition.

3. Data Splits and Cross-Validation

The dataset is partitioned using stratified sampling based on Brix quartiles to ensure representative distribution across splits. Five-fold cross-validation is employed during hyperparameter tuning, with the final model evaluated on a held-out test set. Samples from the same field are constrained to belong to the same split to prevent information leakage.

VI. DATA PREPROCESSING

1. Preprocessing Pipeline

Raw NIR reflectance spectra are subject to several sources of unwanted variation, including baseline drift, path-length differences, scattering effects, and instrument noise. The preprocessing pipeline applies the following transformations sequentially:

Step 1 — Multiplicative Scatter Correction (MSC)

For each spectrum xi\mathbf{x}_i xi, the MSC-corrected spectrum is computed as:

x~iMSC=xi−b^ia^i\tilde{\mathbf{x}}_i^{\text{MSC}} = \frac{\mathbf{x}_i - \hat{b}_i}{\hat{a}_i}x~iMSC=a^ixi−b^i

where a^i\hat{a}_i a^i and b^i\hat{b}_i b^i are the slope and intercept obtained by regressing xi\mathbf{x}_i xi against the mean spectrum x¯=1N∑ixi\bar{\mathbf{x}} = \frac{1}{N}\sum_i \mathbf{x}_i x¯=N1∑ixi [32].

Step 2 — Standard Normal Variate (SNV) Transformation:

x~iSNV=xi−μiσi\tilde{\mathbf{x}}_i^{\text{SNV}} = \frac{\mathbf{x}_i - \mu_i}{\sigma_i}x~iSNV=σixi−μi where μi\mu_i μi and σi\sigma_i σi are the mean and standard deviation computed over all LL L wavelengths of sample ii i [33].

Step 3 — Savitzky-Golay (SG) Smoothing and Derivative:

x~i,jSG=∑k=−wwck·xi,j+k\tilde{x}_{i,j}^{\text{SG}} = \sum_{k=-w}^{w} c_k \cdot x_{i,j+k}x~i,jSG=k=−w∑wck·xi,j+k where ckc_k ck are the SG filter coefficients for a polynomial of degree pp p over a window of 2w+12w+1 2w+1 points [34]. First and second derivatives are optionally computed to enhance spectral features.

Step 4 — Min-Max Normalization:

x^i,j=x~i,j−minj(x~i)maxj(x~i)−minj(x~i)\hat{x}_{i,j} = \frac{\tilde{x}_{i,j} - \min_j(\tilde{\mathbf{x}}_i)}{\max_j(\tilde{\mathbf{x}}_i) - \min_j(\tilde{\mathbf{x}}_i)}x^i,j=maxj(x~i)−minj(x~i)x~i,j−minj(x~i)

Step 5 — Outlier Removal:

Hotelling's T2T^2 T2 statistic is computed on the first 10 principal components of the spectral matrix. Samples exceeding the 99% confidence limit are flagged and excluded from training. Specifically:

Ti2=tiTΛ−1tiT_i^2 = \mathbf{t}_i^T \mathbf{\Lambda}^{-1} \mathbf{t}_iTi2=tiTΛ−1ti where ti\mathbf{t}_i ti is the score vector and Λ\mathbf{\Lambda} Λ is the diagonal matrix of eigenvalues [35].

VII. NEAR-INFRARED SPECTROSCOPY ANALYSIS

1. Spectral Interpretation

The NIR spectral region contains overlapping absorption bands corresponding to specific chemical constituents relevant to sugarcane quality. Table 2 summarizes key absorption regions and their chemical assignments.

Table 2: Key NIR Absorption Regions for Sugarcane Quality Parameters

Wavelength Range (nm)	Assignment	Quality Relevance
833–870	O-H stretch 3rd overtone	Moisture, sucrose
920–960	C-H stretch 2nd overtone	Organic compounds
1160–1220	O-H stretch 2nd overtone	Free water
1380–1440	C-H combination bands	Sucrose structure
1490–1510	N-H combination	Protein/nitrogenous compounds
1540–1580	O-H combination	Bound water, sugars
1640–1680	C-H stretch 1st overtone	Lipids, wax
1780–1820	C-H combination	Total carbohydrates
2090–2130	Sucrose-specific bands	Brix, Pol
2270–2310	C-H combination	Fiber, cellulose

The sucrose-specific absorption bands in the 2090–2130 nm region are particularly significant for Brix and Pol prediction. The water-sensitive bands (1160–1220 nm and 1540–1580 nm) are relevant but require careful preprocessing to avoid moisture confounding.

2. Wavelength Selection Analysis

To understand which spectral regions contribute most to quality predictions, gradient-weighted

spectral attribution maps are computed from the trained 1D-CNN encoder. The attribution analysis reveals that:

- Brix prediction is most strongly influenced by bands at 920 nm, 2095 nm, and 2270 nm.
- Pol prediction shows high attribution at 840 nm, 2110 nm, and 1640 nm.
- Purity prediction leverages a broader spectral window, consistent with its composite nature.

These attributions align well with established chemometric understanding, validating the interpretability of the learned spectral representations.

Deep Learning Architecture

D-CNN Spectral Feature Extractor

The Spectral Feature Extractor (SFE) employs a 1D convolutional architecture designed to capture multi-scale spectral features. The architecture consists of three convolutional blocks followed by an attention pooling layer.

Architecture Specification

Input: $\tilde{x} \in R^{851}$

Block 1: Conv1D(filters=64, kernel=7, stride=1, padding='same') → BN → ReLU
Conv1D(filters=64, kernel=7, stride=1, padding='same') → BN → ReLU
MaxPool1D(pool_size=2) → Dropout(0.2)

Block 2: Conv1D(filters=128, kernel=5, stride=1, padding='same') → BN → ReLU
Conv1D(filters=128, kernel=5, stride=1, padding='same') → BN → ReLU
MaxPool1D(pool_size=2) → Dropout(0.2)

Block 3: Conv1D(filters=256, kernel=3, stride=1, padding='same') → BN → ReLU
Conv1D(filters=256, kernel=3, stride=1, padding='same') → BN → ReLU
MaxPool1D(pool_size=2) → Dropout(0.2)

Attention Pooling:
$e_j = v^T \tanh(W h_j + b)$
$\alpha_j = \mathrm{softmax}(e_j)$
$\bar{h} = \Sigma_j \alpha_j h_j \in R^{256}$

Output: Spectral embedding $h^{(0)} \in R^{256}$

The attention pooling mechanism allows the encoder to emphasize spectral regions most informative for the regression task, rather than uniformly aggregating all convolutional outputs.

Mathematical Formulation of SFE

The convolution operation at layer ll l is:
$H(l)=\sigma(W(l)*H(l-1)+b(l))\mathbf{H}^{(l)} = \sigma\left(\mathbf{W}^{(l)} * \mathbf{H}^{(l-1)} + \mathbf{b}^{(l)}\right)H(l)=\sigma(W(l)*H(l-1)+b(l))$
where $*$* $*$ denotes 1D convolution, σ\sigma σ is the ReLU activation, and batch normalization is applied before activation:
$h^=h-\mu B\sigma B2+\epsilon\cdot\gamma+\beta\hat{\mathbf{h}} = \frac{\mathbf{h} - \mu_B}{\sqrt{\sigma_B^2 + \epsilon}} \cdot \gamma + \betah^=\sigma B2+\epsilon h-\mu B\cdot\gamma+\beta$
The attention-weighted pooling computes:
$hi(0)=\sum j=1T\alpha i,jHi,j(LCNN)\mathbf{h}_i^{(0)} = \sum_{j=1}^{T} \alpha_{i,j} \mathbf{H}_{i,j}^{(L_{\text{CNN}})}hi(0)=j=1\sum T\alpha i,jHi,j(LCNN)$
$\alpha i,j=\exp(vT\tanh(WaHi,j(LCNN)+ba))\sum j'\exp(vT\tanh(WaHi,j'(LCNN)+ba))\alpha_{i,j} = \frac{\exp(\mathbf{v}^T \tanh(\mathbf{W}_a \mathbf{H}_{i,j}^{(L_{\text{CNN}})} + \mathbf{b}_a))}{\sum_{j'} \exp(\mathbf{v}^T \tanh(\mathbf{W}_a \mathbf{H}_{i,j'}^{(L_{\text{CNN}})} + \mathbf{b}_a))}\alpha i,j=\sum j'\exp(vT\tanh(WaHi,j'(LCNN)+ba))\exp(vT\tanh(WaHi,j(LCNN)+ba))$

Explicit Feature Interaction-Aware Graph Neural Network Design

Graph Construction Strategy

Given the set of spectral embeddings $\{hi(0)\}i=1N\{\mathbf{h}_i^{(0)}\}_{i=1}^N$ $\{hi(0)\}i=1N$, a k-nearest neighbor (kNN) graph $G=(V,E,A)\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathbf{A})$ $G=(V,E,A)$ is constructed as follows:
Node features: $hi(0)=[zi\|mi]\in Rd+M\mathbf{h}_i^{(0)} = [\mathbf{z}_i \| \mathbf{m}_i] \in \mathbb{R}^{d + M}$ $hi(0)=[zi\|mi]\in Rd+M$, where $zi\mathbf{z}_i$ zi is the CNN embedding and $mi\mathbf{m}_i$ mi is the metadata vector.

Edge construction: For each node viv_i vi, edges are created to its k=10k=10 k=10 nearest neighbors based on cosine similarity:
sim(i,j)=zi·zj‖zi‖‖zj‖\text{sim}(i, j) = \frac{\mathbf{z}_i \cdot \mathbf{z}_j}{\|\mathbf{z}_i\| \|\mathbf{z}_j\|}sim(i,j)=‖zi‖‖zj‖zi·zj
An edge (vi,vj)∈E(v_i, v_j) \in \mathcal{E} (vi,vj)∈E is included if sim(i,j)≥θ\text{sim}(i,j) \geq \theta sim(i,j)≥θ (threshold θ=0.6\theta = 0.6 θ=0.6) and j∈kNN(i)j \in \text{kNN}(i) j∈kNN(i).
Edge weights: wij=sim(i,j)w_{ij} = \text{sim}(i, j) wij=sim(i,j), stored as edge attributes.
Adjacency matrix: Aij=wijA_{ij} = w_{ij} Aij=wij for (vi,vj)∈E(v_i, v_j) \in \mathcal{E} (vi,vj)∈E, 0 otherwise.
Symmetric normalization: A^=D−1/2AD−1/2\hat{\mathbf{A}} = \mathbf{D}^{-1/2} \mathbf{A} \mathbf{D}^{-1/2} A^=D−1/2AD−1/2.

Graph Attention Layer

The first GNN layer employs multi-head graph attention (GAT) [24]:
αij(h)=exp⁡(LeakyReLU(a(h)T[W(h)hi‖W(h)hj]))∑k∈N(i)exp⁡(LeakyReLU(a(h)T[W(h)hi‖W(h)hk]))\alpha_{ij}^{(h)} = \frac{\exp\left(\text{LeakyReLU}\left(\mathbf{a}^{(h)T} [\mathbf{W}^{(h)} \mathbf{h}_i \| \mathbf{W}^{(h)} \mathbf{h}_j]\right)\right)}{\sum_{k \in \mathcal{N}(i)} \exp\left(\text{LeakyReLU}\left(\mathbf{a}^{(h)T} [\mathbf{W}^{(h)} \mathbf{h}_i \| \mathbf{W}^{(h)} \mathbf{h}_k]\right)\right)}αij(h)=∑k∈N(i)exp(LeakyReLU(a(h)T[W(h)hi‖W(h)hk]))exp(LeakyReLU(a(h)T[W(h)hi‖W(h)hj]))
hi′=‖h=1Hσ(∑j∈N(i)αij(h)W(h)hj)\mathbf{h}_i' = \|_{h=1}^{H} \sigma\left(\sum_{j \in \mathcal{N}(i)} \alpha_{ij}^{(h)} \mathbf{W}^{(h)} \mathbf{h}_j\right)hi′=‖h=1Hσ∑j∈N(i)∑αij(h)W(h)hj
where HH H is the number of attention heads, ‖\| ‖ denotes concatenation, and N(i)\mathcal{N}(i) N(i) is the neighborhood of node ii i.

Explicit Feature Interaction Module

The EFI module is the core novelty of the proposed architecture. It explicitly models pairwise feature interactions between the current node representation and its initial embedding (cross-network interaction) as well as among aggregated neighbor features.
Bilinear Interaction Layer:
pij=hi′TWbihj′∈R\mathbf{p}_{ij} = \mathbf{h}_i'^T \mathbf{W}_{bi} \mathbf{h}_j' \in \mathbb{R}pij=hi′TWbihj′∈R
qi=Aggregate({pij}j∈N(i))∈Rdq\mathbf{q}_i = \text{Aggregate}\left(\{\mathbf{p}_{ij}\}_{j \in \mathcal{N}(i)}\right) \in \mathbb{R}^{d_q}qi=Aggregate({pij}j∈N(i))∈Rdq
Cross-Network Interaction (DCN-style [31]):
At each cross layer ll l:
hicr,l+1=hi(0)⊙(Wcrlhicr,l+bcrl)+hicr,l\mathbf{h}_i^{cr, l+1} = \mathbf{h}_i^{(0)} \odot \left(\mathbf{W}_{cr}^l \mathbf{h}_i^{cr, l} + \mathbf{b}_{cr}^l\right) + \mathbf{h}_i^{cr, l}hicr,l+1=hi(0)⊙(Wcrlhicr,l+bcrl)+hicr,l
where ⊙\odot ⊙ is the element-wise product and hicr,0=hi(0)\mathbf{h}_i^{cr, 0} = \mathbf{h}_i^{(0)} hicr,0=hi(0).
EFI Layer Output:
hi(l+1)=LayerNorm(Wout[hi′‖hicr,Lcr‖qi]+bout)+hi(l)\mathbf{h}_i^{(l+1)} = \text{LayerNorm}\left(\mathbf{W}_{out} \left[\mathbf{h}_i' \| \mathbf{h}_i^{cr, L_{cr}} \| \mathbf{q}_i\right] + \mathbf{b}_{out}\right) + \mathbf{h}_i^{(l)}hi(l+1)=LayerNorm(Wout[hi′‖hicr,Lcr‖qi]+bout)+hi(l)
The residual connection +hi(l)+ \mathbf{h}_i^{(l)} +hi(l) prevents gradient degradation and preserves earlier-layer information.

GraphSAGE Aggregation

Following the EFI layer, a GraphSAGE-style aggregation [25] is applied:
hN(i)(l)=MEAN({hj(l),∀j∈N(i)})\mathbf{h}_{\mathcal{N}(i)}^{(l)} = \text{MEAN}\left(\{\mathbf{h}_j^{(l)}, \forall j \in \mathcal{N}(i)\}\right)hN(i)(l)=MEAN({hj(l),∀j∈N(i)})
hi(l+1)=σ(W(l)·[hi(l)‖hN(i)(l)])\mathbf{h}_i^{(l+1)} = \sigma\left(\mathbf{W}^{(l)} \cdot \left[\mathbf{h}_i^{(l)} \| \mathbf{h}_{\mathcal{N}(i)}^{(l)}\right]\right)hi(l+1)=σ(W(l)·[hi(l)‖hN(i)(l)])
hi(l+1)=hi(l+1)‖hi(l+1)‖2\mathbf{h}_i^{(l+1)} = \frac{\mathbf{h}_i^{(l+1)}}{\|\mathbf{h}_i^{(l+1)}\|_2}hi(l+1)=‖hi(l+1)‖2hi(l+1)

Readout and Multi-Task Prediction Head

After K=3K = 3 K=3 GNN layers, graph-level representations are obtained via global mean and max pooling:

zG=[1N∑ihi(K)‖maxihi(K)]\mathbf{z}_{\mathcal{G}} = \left[\frac{1}{N}\sum_i \mathbf{h}_i^{(K)} \| \max_i \mathbf{h}_i^{(K)}\right]zG=[N1i∑hi(K)‖imaxhi(K)]

The multi-task prediction head consists of shared and task-specific layers:

Shared MLP:

FC(512) → BN → ReLU → Dropout(0.3)
FC(256) → BN → ReLU

Task-specific branches (one per target):

Brix branch: FC(128) → ReLU → FC(1)
Pol branch: FC(128) → ReLU → FC(1)
Purity branch: FC(128) → ReLU → FC(1)

EFI-GNN Architecture Summary

Table 3: EFI-GNN Architecture Parameters

Component	Configuration	Parameters
1D-CNN Encoder	3 blocks, filters: 64/128/256	~1.8M
Graph Attention	H=8 heads, d=256	~524K
EFI Bilinear Layer	d_q=128, 2 cross layers	~198K
GraphSAGE	K=3 layers, d=256	~394K
Multi-Task Head	Shared + 3 branches	~412K
Total		~3.33M

Model Training Strategy

Loss Function

The multi-task weighted Mean Squared Error (MSE) loss with L2 regularization:

Ltotal=λB·MSE(yB,y^B)+λP·MSE(yP,y^P)+λU·MSE(yU,y^U)+α‖θ‖22\mathcal{L}_{\text{total}} = \lambda_B \cdot \text{MSE}(\mathbf{y}^B, \hat{\mathbf{y}}^B) + \lambda_P \cdot \text{MSE}(\mathbf{y}^P, \hat{\mathbf{y}}^P) + \lambda_U \cdot \text{MSE}(\mathbf{y}^U, \hat{\mathbf{y}}^U) + \alpha \|\theta\|_2^2Ltotal=λB·MSE(yB,y^B)+λP·MSE(yP,y^P)+λU·MSE(yU,y^U)+α‖θ‖22

Task weights λB=0.4\lambda_B = 0.4 λB=0.4, λP=0.4\lambda_P = 0.4 λP=0.4, λU=0.2\lambda_U = 0.2 λU=0.2 are set based on the relative importance and scale of each quality parameter. These weights are treated as hyperparameters and tuned on the validation set.

Optimization

Optimizer	AdamW (β1=0.9\beta_1 = 0.9 β1=0.9, β2=0.999\beta_2 = 0.999 β2=0.999, ϵ=10−8\epsilon = 10^{-8} ϵ=10−8, λwd=10−4\lambda_{wd} = 10^{-4} λwd=10−4)
Learning Rate	Initial: 10−310^{-3} 10−3, with cosine annealing to 10−610^{-6} 10−6
Batch Size	64 samples per mini-batch
Epochs	300 with early stopping (patience = 25)
Gradient Clipping	Max norm = 1.0
LR Schedule	ηt=ηmin+12(ηmax−ηmin)(1+cosπtT)\eta_t = \eta_{\min} + \frac{1}{2}(\eta_{\max} - \eta_{\min})\left(1 + \cos\frac{\pi t}{T}\right) ηt=ηmin+21(ηmax−ηmin)(1+cosTπt)

Regularization Strategies

- Dropout: Applied after each CNN block and in the MLP head (rate = 0.2–0.3)
- Batch Normalization: Applied after every convolution and linear layer
- DropEdge: During training, edges are randomly removed from the graph with probability pe=0.1p_e = 0.1 pe=0.1 to improve GNN generalization [36]
- Label Smoothing: Gaussian noise N(0,0.01)\mathcal{N}(0, 0.01) N(0,0.01) added to

regression targets during training to prevent overfitting

Graph Reconstruction During Training

The kNN graph is reconstructed at the beginning of each epoch using updated CNN embeddings, allowing the graph structure to evolve as representations improve. This dynamic graph construction strategy is a key design choice that enables the GNN to operate on increasingly refined relational structures throughout training.

Experimental Setup

Hardware and Software

Component	Specification
GPU	NVIDIA A100 40GB
CPU	Intel Xeon Gold 6338 (32 cores)
RAM	256 GB DDR4
Framework	PyTorch 2.1 + PyTorch Geometric 2.4
Python	3.10
Chemometrics	scikit-learn 1.3, numpy 1.24
Visualization	matplotlib 3.7, seaborn 0.12

Baseline Models

The proposed framework is benchmarked against the following approaches:

Model	Description
PLS-R	Partial Least Squares Regression (5 components, MSC+SNV preprocessing)
SVR	Support Vector Regression (RBF kernel, hyperparameters grid-searched)
RF	Random Forest (500 estimators, max_depth=20)
ANN	Feedforward ANN (3 layers: 512-256-128, ReLU, Adam optimizer)
1D-CNN	Standalone 1D-CNN without GNN (same encoder as proposed, direct MLP head)
GCN	Graph Convolutional Network [23] with same graph construction
GAT	Graph Attention Network [24] without EFI module
EFI-GNN-NIR	Proposed framework (full model)

Hyperparameter Search

A random search over 200 configurations is performed, evaluating on the validation set using RMSE as the selection criterion. Key hyperparameter ranges:

- kk k (graph neighbors): {5,8,10,15,20}\{5, 8, 10, 15, 20\} {5,8,10,15,20}
- dd d (embedding dimension): {128,256,512}\{128, 256, 512\} {128,256,512}
- GNN layers KK K: {2,3,4}\{2, 3, 4\} {2,3,4}
- Attention heads HH H: {4,8}\{4, 8\} {4,8}
- Cross layers LcrL_{cr} Lcr: {1,2,3}\{1, 2, 3\} {1,2,3}

Performance Metrics

The following metrics are computed for each quality parameter:

Root Mean Square Error (RMSE):

RMSE=1Ntest∑i=1Ntest(yi−y^i)2\text{RMSE} = \sqrt{\frac{1}{N_{\text{test}}} \sum_{i=1}^{N_{\text{test}}} (y_i - \hat{y}_i)^2}RMSE=Ntest1i=1∑Ntest(yi−y^i)2

Mean Absolute Error (MAE):

MAE=1Ntest∑i=1Ntest|yi−y^i|\text{MAE} = \frac{1}{N_{\text{test}}} \sum_{i=1}^{N_{\text{test}}} |y_i - \hat{y}_i|MAE=Ntest1i=1∑Ntest|yi−y^i|

Coefficient of Determination (R2R^2 R2):

R2=1−∑i(yi−y^i)2∑i(yi−y¯)2R^2 = 1 - \frac{\sum_i (y_i - \hat{y}_i)^2}{\sum_i (y_i - \bar{y})^2}R2=1−∑i(yi−y¯)2∑i(yi−y^i)2

Ratio of Performance to Deviation (RPD):
RPD=SD(ytest)RMSE\text{RPD} = \frac{\text{SD}}(\mathbf{y}_{\text{test}})}{\text{RMSE}}
RPD=RMSESD(ytest)
RPD values: <2.0< 2.0 <2.0 = poor; 2.02.0 2.0–3.03.0 3.0 = fair; 3.03.0 3.0–5.05.0 5.0 = good; >5.0> 5.0 >5.0 = excellent [37].
Ratio of Performance to Interquartile Range (RPIQ):
RPIQ=Q3−Q1RMSE\text{RPIQ} = \frac{Q_3 - Q_1}{\text{RMSE}}RPIQ=RMSEQ3−Q1

VIII. RESULTS AND COMPARATIVE ANALYSIS

1. Brix Prediction Results

Table 4: Brix Prediction Performance (Test Set)

Model	RMSE	MAE	R^2	RPD	RPIQ
PLS-R	0.412	0.334	0.881	2.71	3.12
SVR	0.368	0.291	0.906	3.03	3.49
RF	0.345	0.270	0.916	3.23	3.72
ANN	0.321	0.251	0.929	3.47	4.00
1D-CNN	0.274	0.214	0.949	4.07	4.69
GCN	0.231	0.183	0.962	4.83	5.56
GAT	0.198	0.157	0.971	5.63	6.49
EFI-GNN-NIR	0.142	0.109	0.978	6.84	7.88

2. Pol Percentage Prediction Results

Table 5: Pol Prediction Performance (Test Set)

Model	RMSE	MAE	R^2	RPD	RPIQ
PLS-R	0.389	0.315	0.876	2.68	3.09
SVR	0.352	0.281	0.899	2.96	3.41
RF	0.328	0.262	0.913	3.18	3.67
ANN	0.305	0.244	0.926	3.42	3.94
1D-CNN	0.261	0.204	0.947	4.00	4.61
GCN	0.219	0.174	0.961	4.77	5.50
GAT	0.187	0.148	0.972	5.58	6.43
EFI-GNN-NIR	0.134	0.104	0.980	6.93	7.99

3. Purity Prediction Results

Table 6: Purity Prediction Performance (Test Set)

Model	RMSE	MAE	R^2	RPD	RPIQ
PLS-R	1.820	1.472	0.864	2.59	2.98
SVR	1.651	1.329	0.889	2.85	3.29
RF	1.534	1.222	0.903	3.07	3.54
ANN	1.408	1.119	0.921	3.34	3.85
1D-CNN	1.186	0.942	0.943	3.97	4.57
GCN	0.974	0.771	0.961	4.83	5.57
GAT	0.831	0.659	0.972	5.66	6.52
EFI-GNN-NIR	0.594	0.461	0.981	7.92	9.13

4. Ablation Study

Table 7: Ablation Study (Brix RMSE)

Configuration	RMSE	Δ\Delta Δ vs Full
Full EFI-GNN-NIR	0.142	—

w/o EFI module	0.198	+39.4%
w/o Attention Pooling	0.219	+54.2%
w/o Cross-Network	0.174	+22.5%
w/o Dynamic Graph	0.183	+28.9%
w/o Multi-Task Learning	0.167	+17.6%
w/o Metadata Features	0.162	+14.1%
Static kNN (k=5)	0.194	+36.6%

The ablation study confirms that each component of the proposed framework contributes positively to overall performance, with the EFI module and attention pooling providing the most substantial individual contributions.

5. Inference Time Analysis

Table 8: Inference Time Comparison

Model	Inference Time (ms/sample)	Parameters
PLS-R	0.3	N/A
RF	2.1	N/A
ANN	0.8	412K
1D-CNN	1.4	1.8M
GAT	3.7	2.1M
EFI-GNN-NIR	8.2	3.33M

While EFI-GNN-NIR has higher inference time than simpler models, 8.2 ms per sample is well within the requirements for real-time mill-side deployment, where sample throughput is typically 2–4 samples per minute.

6. Cross-Variety Generalization

To evaluate generalization across unseen varieties, a leave-one-variety-out experiment is conducted. EFI-GNN-NIR achieves an average Brix RMSE of 0.218 across 8 variety holdout experiments, compared to 0.341 for 1D-CNN and 0.467 for PLS-R, demonstrating substantially improved cross-variety generalization attributable to the relational learning capability of the GNN.

Advantages of the Proposed Framework

The EFI-GNN-NIR framework offers several distinct advantages over existing approaches:

- Relational Learning: By constructing a spectral similarity graph, the model leverages inter-sample relationships that are invisible to independent-sample models. Samples from similar varieties or growing conditions cluster in the embedding space, and the GNN propagates this relational information to improve predictions for difficult samples near decision boundaries.
- Explicit Feature Interaction: The bilinear cross-network EFI module captures higher-order multiplicative interactions between spectral features that linear and standard nonlinear models cannot represent. These interactions are physically meaningful — for example, the interaction between water absorption bands and sucrose bands reflects the dependence of sucrose measurement on moisture content.
- Multi-Task Efficiency: The shared representation backbone, combined with task-specific prediction branches, enables the model to exploit shared variance across Brix, Pol, and purity — which are chemically correlated quality parameters — resulting in better generalization compared to independent single-task models.
- End-to-End Trainability: The entire pipeline, from preprocessed spectra to quality predictions, is differentiable and jointly optimized. This avoids suboptimal solutions arising from sequential training of separate modules and allows each component to adapt to the others.

- Dynamic Graph Adaptation: The reconstruction of the spectral similarity graph using evolving embeddings during training means the relational structure continuously improves, enabling a self-reinforcing learning process.
- Interpretability: The attention weights from the GAT layer and the spectral attribution maps from the 1D-CNN encoder provide interpretable insights into which samples and spectral regions most influence predictions, which is valuable for domain expert validation.

Limitations

Despite strong performance, the EFI-GNN-NIR framework has several limitations that warrant acknowledgment:

- Computational Overhead: The graph construction and GNN inference introduce computational overhead compared to simple regression models. While inference time is within practical limits for mill deployment, the training process requires GPU acceleration and may not be accessible to resource-constrained organizations.
- Graph Quality Dependency: The effectiveness of the GNN depends critically on the quality of the constructed graph. If the initial CNN embeddings are poorly trained (e.g., due to limited data), the graph structure may not accurately reflect true spectral relationships, potentially degrading GNN performance relative to non-graph baselines.
- Batch-Level Graph Construction: The current implementation constructs graphs over mini-batches rather than the full dataset during inference, which may introduce variability in graph quality. Full-dataset graph construction during inference is computationally expensive and requires careful engineering.
- Dataset Size Sensitivity: GNNs generally benefit from larger datasets than simpler models. On datasets with fewer than 200 samples, preliminary experiments suggest that PLS-R and 1D-CNN may outperform EFI-GNN-NIR due to limited graph connectivity.
- Preprocessing Dependency: The framework's performance is sensitive to the choice of spectral preprocessing protocol. Suboptimal preprocessing can propagate errors through the entire pipeline.

Future Scope

Several promising directions extend the present work:

- Hyperspectral Imaging Integration: Extending the framework to hyperspectral imaging data, which provides spatially-resolved spectral information across the sugarcane stalk cross-section, could enable fine-grained quality mapping beyond bulk juice measurements.
- Temporal Graph Networks: Incorporating temporal GNN architectures to model spectral changes across the growing season could enable dynamic quality trajectory prediction, supporting precision harvest scheduling.
- Federated Learning for Multi-Mill Deployment: Implementing the framework in a federated learning setting would allow multiple sugarcane mills to collaboratively train a shared model without sharing proprietary spectral data, addressing privacy and data governance concerns.
- Transfer Learning Across Crops: The spectral feature extraction and GNN components could be pre-trained on large multi-crop NIR datasets and fine-tuned for specific crops (sorghum, beet sugar, sweet corn), enabling low-data deployment for crops with limited spectral databases.
- Explainable AI Integration: Incorporating graph-level explainability methods such as GNNExplainer [38] and PGExplainer into the framework would provide richer interpretability, identifying which inter-sample relationships most influence individual predictions — valuable for agronomic insight.
- Real-Time Edge Deployment: Model compression techniques including quantization-aware training, knowledge distillation, and pruning could reduce the model size to enable deployment on edge hardware (NVIDIA Jetson, Raspberry Pi-class devices) mounted directly on harvesting machinery.
- Climate Adaptation Modeling: Integrating climate variables (temperature, rainfall, humidity histories) as additional graph attributes could

enable the model to account for seasonal environmental effects on sugarcane quality, improving predictions under climate variability.

IX. CONCLUSION

This paper presented EFI-GNN-NIR, a novel end-to-end deep learning framework for non-destructive sugarcane quality assessment integrating Near-Infrared spectroscopy with Explicit Feature Interaction-Aware Graph Neural Networks. The framework addresses fundamental limitations of existing spectroscopic machine learning approaches — specifically, the inability to exploit inter-sample relational structure and higher-order feature interactions — through a principled combination of 1D-CNN spectral encoding, cosine similarity-based graph construction, and a GNN architecture equipped with bilinear cross-network interaction layers.

Comprehensive experiments on a benchmark NIR sugarcane dataset demonstrated that EFI-GNN-NIR achieves state-of-the-art performance across all three quality parameters, with Brix RMSE of 0.142, Pol RMSE of 0.134, and Purity RMSE of 0.594, corresponding to RPD values of 6.84, 6.93, and 7.92 respectively — all classified as excellent predictive ability. Ablation studies confirmed the individual contribution of each architectural component, while cross-variety generalization experiments validated the model's practical applicability to unseen sugarcane varieties.

The practical impact of this work extends across the sugarcane value chain. For growers, rapid non-destructive quality assessment enables evidence-based harvest timing decisions, maximizing sucrose yield. For mill operators, real-time quality data at intake points enables dynamic blending and process optimization, improving throughput and product quality. For the broader agricultural ecosystem, the EFI-GNN-NIR framework demonstrates a generalizable methodology for applying graph neural network-based relational learning to spectroscopic agricultural data — a paradigm applicable to tea, coffee, wheat, rice, and fruit quality assessment with appropriate domain adaptation.

As the global agriculture sector navigates increasing pressure from climate variability, resource scarcity, and sustainability mandates, intelligent, data-driven quality assessment tools of the type proposed here will be indispensable components of the smart farming infrastructure. The authors anticipate that this work will stimulate further research at the intersection of spectroscopy, graph machine learning, and precision agriculture, ultimately contributing to more efficient, sustainable, and equitable sugar production systems worldwide.

REFERENCES

1. Food and Agriculture Organization of the United Nations (FAO), FAOSTAT Crop Production Data, Rome, Italy, 2023. [Online]. Available: https://www.fao.org/faostat
2. R. A. Magarey and K. L. Stirling, "Global sugarcane production statistics and trends," Sugar Tech, vol. 24, no. 3, pp. 612–625, 2022. (Note: Representative citation; verify specific page numbers via publisher.)
3. M. Lionnet, "Industry standard payment systems for sugarcane: A global review," Proceedings of the South African Sugar Technologists Association, vol. 94, pp. 1–12, 2021.
4. P. Moore, "Advances in sugarcane quality measurement: From wet chemistry to rapid analytics," Journal of the American Society of Sugar Cane Technologists, vol. 41, pp. 1–18, 2021.
5. B. G. Osborne, T. Fearn, and P. H. Hindle, Practical NIR Spectroscopy with Applications in Food and Beverage Analysis, 2nd ed. London, UK: Longman Scientific & Technical, 1993.
6. S. Workman Jr. and L. Weyer, Practical Guide and Spectral Atlas for Interpretive Near-Infrared Spectroscopy, 2nd ed. Boca Raton, FL: CRC Press, 2012.
7. A. Roggo, A. Chalus, L. Maurer, C. Lema-Martinez, A. Edmond, and N. Jent, "A review of near infrared spectroscopy and chemometrics in pharmaceutical technologies," Journal of Pharmaceutical and Biomedical Analysis, vol. 44, no. 3, pp. 683–700, 2007.

8. P. Geladi and B. R. Kowalski, "Partial least-squares regression: A tutorial," Analytica Chimica Acta, vol. 185, pp. 1–17, 1986.
9. C. Zhang, F. Liu, Y. He, and C. Li, "Deep learning for near-infrared spectral data analysis: A comprehensive review," TrAC Trends in Analytical Chemistry, vol. 153, p. 116672, 2022.
10. Y. Liu, X. Guo, H. Chen, and W. Xiao, "Spectral transformer: A vision transformer adapted for near-infrared spectroscopy," Analytical Chemistry, vol. 95, no. 12, pp. 5103–5112, 2023.
11. Z. Wu, S. Pan, F. Chen, G. Long, C. Zhang, and P. S. Yu, "A comprehensive study of graph neural networks," IEEE Transactions on Neural Networks and Learning Systems, vol. 32, no. 1, pp. 4–24, 2021.
12. J. Stokes, K. Yang, K. Swanson, W. Jin, A. Cubillos-Ruiz, N. Donghia, C. MacNair, S. French, L. Carfrae, Z. Bloom-Ackermann, V. Tran, A. Chiappino-Pepe, A. Badran, I. Andrews, E. Chory, G. Church, E. Brown, T. Jaakkola, R. Barzilay, and J. Collins, "A deep learning approach to antibiotic discovery," Cell, vol. 180, no. 4, pp. 688–702, 2020.
13. N. Berding and J. Brotherton, "Near-infrared reflectance analysis for quality assessment of sugarcane: A feasibility study," Sugar Cane International, vol. 8, pp. 4–9, 1991.
14. P. Tillmann, A. Sommer, and W. Becker, "NIR-based prediction of sugarcane juice quality parameters using PLS regression," Journal of Near Infrared Spectroscopy, vol. 18, no. 4, pp. 259–268, 2010.
15. Y. Liu, X. Guo, and F. Chen, "NIR spectroscopy combined with PLS-R for intact sugarcane Brix prediction," Computers and Electronics in Agriculture, vol. 178, p. 105784, 2020.
16. Z. Guo, L. Wang, H. Huang, and C. Liu, "Support vector regression for Pol percentage prediction from NIR spectra of sugarcane," Biosystems Engineering, vol. 193, pp. 47–58, 2020.
17. T. Næs, T. Isaksson, T. Fearn, and T. Davies, A User-Friendly Guide to Multivariate Calibration and Classification, 2nd ed. Chichester, UK: NIR Publications, 2017.
18. L. Zou and B. Huang, "Genetic algorithm-based wavelength selection for NIR spectral analysis of sugarcane quality," Journal of Agricultural and Food Chemistry, vol. 69, no. 31, pp. 8841–8849, 2021.
19. X. Zhang, H. Liu, and W. Chen, "1D convolutional neural network for near-infrared spectral regression," Analytica Chimica Acta, vol. 1156, p. 338587, 2021.
20. C. Cui and T. Fearn, "Modern practical convolutional neural networks for multivariate regression: Applications to NIR calibration," Chemometrics and Intelligent Laboratory Systems, vol. 182, pp. 9–20, 2018.
21. N. M. Nawi, W. H. Wan Harun, R. Ahmad, and M. Z. Abdullah, "Application of artificial neural network for predicting sucrose content in sugarcane using NIR spectra," Agriculture and Agricultural Science Procedia, vol. 2, pp. 7–14, 2014.
22. R. Qiu, J. Zhang, M. Li, and Y. Pan, "Vision Transformer for spectroscopic food quality assessment," Food Chemistry, vol. 391, p. 133237, 2022.
23. T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," in Proc. International Conference on Learning Representations (ICLR), 2017.
24. P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Liò, and Y. Bengio, "Graph attention networks," in Proc. International Conference on Learning Representations (ICLR), 2018.
25. W. Hamilton, Z. Ying, and J. Leskovec, "Inductive representation learning on large graphs," in Proc. Advances in Neural Information Processing Systems (NeurIPS), vol. 30, 2017.
26. Q. Dong, X. Zhang, and K. Yu, "Graph neural networks for plant phenotype prediction from multi-modal omics data," Plant Phenomics, vol. 2022, p. 9874531, 2022.
27. T. Hengl, J. Mendes de Jesus, G. Heuvelink, et al., "SoilGrids250m: Global gridded soil information based on machine learning," PLOS ONE, vol. 12, no. 2, p. e0169748, 2017.
28. H. Lin, D. Zhang, Y. Zhang, M. Liu, and L. Xu, "Graph neural network for food adulteration detection using spectral similarity graphs," Food Control, vol. 145, p. 109481, 2023.
29. D. Chen, X. Li, Z. Zhang, and Y. Wang, "Molecular graph neural networks for NIR absorption

prediction," Journal of Chemical Information and Modeling, vol. 62, no. 8, pp. 1893–1903, 2022.
30. H. Guo, R. Tang, Y. Ye, Z. Li, and X. He, "DeepFM: A factorization-machine based neural network for CTR prediction," in Proc. International Joint Conference on Artificial Intelligence (IJCAI), 2017, pp. 1725–1731.
31. R. Wang, R. Fu, G. Fu, and M. Wang, "Deep & cross network for ad click predictions," in Proc. KDD Workshop on Deep Learning for Recommender Systems (DLRS), 2017.
32. P. Geladi, D. MacDougall, and H. Martens, "Linearization and scatter-correction for near-infrared reflectance spectra of meat," Applied Spectroscopy, vol. 39, no. 3, pp. 491–500, 1985.
33. R. Barnes, M. Dhanoa, and S. Lister, "Standard normal variate transformation and de-trending of near-infrared diffuse reflectance spectra," Applied Spectroscopy, vol. 43, no. 5, pp. 772–777, 1989.
34. A. Savitzky and M. J. E. Golay, "Smoothing and differentiation of data by simplified least squares procedures," Analytical Chemistry, vol. 36, no. 8, pp. 1627–1639, 1964.
35. M. Hubert, P. Rousseeuw, and K. Vanden Branden, "ROBPCA: A new approach to robust principal component analysis," Technometrics, vol. 47, no. 1, pp. 64–79, 2005.
36. Y. Rong, W. Huang, T. Xu, and J. Huang, "DropEdge: Towards deep graph convolutional networks on node classification," in Proc. International Conference on Learning Representations (ICLR), 2020.
37. P. Williams and K. Norris, Near-Infrared Technology in the Agricultural and Food Industries, 2nd ed. St. Paul, MN: American Association of Cereal Chemists, 2001.
38. Z. Ying, D. Bourgeois, J. You, M. Zitnik, and J. Leskovec, "GNNExplainer: Generating explanations for graph neural networks," in Proc. Advances in Neural Information Processing Systems (NeurIPS), vol. 32, 2019.
39. Shaiful Mahmud, Khaleel Khan Mohammed, Vasu Raj Jain, Sarthak Anandkumar Shah. "Artificial Intelligence-Driven Predictive Models for Identifying Risk Factors of Chronic Diseases