Integrating Machine Learning and Deep Learning for Enhanced Soil Fertility Prediction and Crop Recommendation in Precision Agriculture

Ling Siew Win
School of Computing
Asia Pacific University of Technology and Innovation (APU)
Kuala Lumpur, Malaysia
tp072643@mail.apu.edu.my

Aryssa Goh May Lynn
School of Computing
Asia Pacific University of Technology and Innovation (APU)
Kuala Lumpur, Malaysia
tp072774@mail.apu.edu.my

Hussain Moizbhai Halderwala
School of Computing
Asia Pacific University of Technology and Innovation (APU)
Kuala Lumpur, Malaysia
tp066628@mail.apu.edu.my

Jinendra Muraly
School of Computing
Asia Pacific University of Technology and Innovation (APU)
Kuala Lumpur, Malaysia
tp072133@mail.apu.edu.my

Adeline Sneha John Chrisastum
School of Computing
Asia Pacific University of Technology and Innovation (APU)
Kuala Lumpur, Malaysia
adeline.john@apu.edu.my

Juhairi Aris Muhamad Shuhili
School of Engineering
Asia Pacific University of Technology and Innovation (APU)
Kuala Lumpur, Malaysia
juhairi.shuhili@apu.edu.my

***Abstract*— In this research, soil fertility prediction system is proposed to increase the efficiency of yield production and vegetation cover. Based on the nitrogen (NO3), phosphorous (P), and potassium (K) levels in the dataset, our method combines machine learning and deep learning techniques to identify soil nutrients and suggest ideal crops. We combine and preprocess datasets, and then use different deep learning architectures and regression models, such as MLP Regressor, Linear Regressor, Random Forest Regressor to precisely estimate vegetation cover. Our findings show that when it comes to accuracy and error measures, the MLP Regressor performs the best. In summary, this study provides a significant understanding of soil nutrient analysis and crop suggestion for environmentally friendly farming methods.**

Keywords*— *soil fertility prediction, machine learning, deep learning, MLP Regressor, NPK value

1.0 INTRODUCTION

Literature Review on:

(i) Similar Projects

Machine learning (ML) and deep learning (DL) algorithms have emerged as powerful tools for soil fertility prediction, leveraging their capability to analyze complex datasets and extract meaningful insights. Recently, there has been significant progress in forecasting soil fertility features using machine learning (ML) methods including gradient boosting regression (GBR), random forest (RF), and support vector machine (SVM). For instance, (Zhang et al., 2018) utilized SVM and RF algorithms to accurately predict the organic carbon content in soil, while (Wang et al., 2020) achieved superior prediction performance for soil nitrogen levels using GBR. These findings underscore the potential of ML techniques in estimating soil fertility levels by leveraging historical soil data, weather patterns, and crop outcomes, thereby offering valuable insights for precision agriculture management.

In addition to ML, DL algorithms, particularly convolutional neural networks (CNNs) and recurrent neural networks (RNNs), have shown significant promise in soil analysis tasks. (Zhang and Wang, 2019) surpassed previous methodologies by developing a CNN-based model for soil texture categorization using spectral reflectance data, highlighting DL's effectiveness in feature extraction and classification tasks. Furthermore, (Li et al., 2021) utilized RNNs to capture spatiotemporal dependencies in soil moisture dynamics, leading to improved temporal soil moisture prediction. These studies illustrate how DL techniques excel in uncovering intricate correlations within soil data, enabling more precise analysis and prediction of soil characteristics for enhanced agricultural practices.

ML and DL algorithms play a crucial role in recommending suitable crops based on soil properties, climate conditions, and historical data. (Patel et al., 2019) developed a hybrid recommendation system integrating ML and DL techniques, while (Gupta et al., 2020) employed DL for accurate crop selection. These studies highlight the importance of advanced algorithms in optimizing crop choices for enhanced agricultural productivity.

(ii) Methodology/Approach

The system classifies nutrients in soil primarily using machine learning (ML) and deep learning (DL) techniques. The first step is data preparation, which involves collecting pertinent features from the information such as soil nutrients like nitrogen (N), phosphorus (P), and potassium (K).

One of the preprocessing steps include handling missing values and normalizing the data for uniformity across features. The dataset is then divided into testing and training sets in order to train and assess the model. Using the training data, a variety of machine learning and deep learning models, including Support Vector Machines (SVM), Decision Trees, Random Forests, Neural Networks, etc., are developed.

To find the best-performing model for soil nutrient classification, model evaluation entails evaluating the mean root square error, accuracy, precision, R2-score, and absolute error on the testing data. Techniques for cross-validation guarantee the generalisation and dependability of the chosen model. The methodology offers insights on crop recommendation based on the soil nutrient data presented in the dataset, even though the major focus is still on soil nutrient classification.

(iii) Conclusion/Recommendation

In conclusion, the literature review underscores the effectiveness of machine learning (ML) and deep learning (DL) algorithms in soil fertility prediction and nutrient classification. ML techniques like SVM, RF, and GBR, along with DL methods such as CNNs and RNNs, show promise in accurately estimating soil characteristics and recommending suitable crops based on soil properties and climate data.

Methodologically, the approach involves rigorous data preparation and model training, with evaluation metrics ensuring model reliability. The integration of ML and DL holds significant potential for optimizing agricultural practices and enhancing productivity. Future research may focus on refining models and deploying them in real-world agricultural settings for sustainable farming practices.

2.0 MATERIALS AND METHODS

2.1 Abbreviations and Acronyms

(i) Purpose

The purpose of this research is to analyse the soil fertility based on the soil elements within a given vegetation coverage dataset through implementing Machine Learning and recommend the most appropriate crop based on the NO3, P, K value of the dataset, as N, P, and K are the major elements for the crop. In this research, NO3 was utilized as N in the merged dataset.

Several models that are used are Linear Model, Support Vector Regression (Linear, RBF, Poly), Decision Tree Regression, and Random Forest Regression. Soil fertility analysis is done through the stated machine learning models which were utilized to predict the vegetation coverage based on the soil elements that are present in the dataset.

In addition to the models, three different approaches which are MLRegressor, one of the Machine Learning methods, and two other types of Deep Learning method; Long Short-Term Memory (LSTM) and Simple Recurrent Neural Network (SimpleRNN) were also implemented to find the best performance in terms of accuracy and efficiency of Machine Learning and Deep Learning which suits for soil nutrients analysis and crop recommendation.

By exploring different approaches, this research aims to identify the most effective approach for soil nutrient analysis and crop recommendation.

(ii) Parameters

The features (X) for soil nutrient coverage prediction include micronutrients like zinc (Zn), copper (Cu), iron (Fe), calcium (Ca), magnesium (Mg), and sodium (Na), as well as various soil nutrient attributes like nitrate (NO3), ammonium (NH4), phosphorus (P), potassium (K), sulphate (SO4), boron (B), organic matter, and pH level. The vegetation cover, which is a measure of soil fertility and overall ecosystem health, is the target variable (Y) to be predicted.

In contrast, the attributes (X) for crop recommendation include geographical coordinates denoted by latitude and longitude as well as the nutritional requirements for crops, such as nitrogen (N), phosphorus (P), and potassium (K). The label that suggests a crop based on the provided nutrient requirements and geographic location is the target variable (Y). In agricultural contexts, these factors offer crucial insights for comprehending and simulating crop suitability and soil fertility.

3.0 RESULTS AND DISCUSSION

3.1 Discussion

Nitrogen, crucial for chlorophyll and protein synthesis, is essential for plant vitality. Phosphorus supports cell division, energy transfer, and robust root growth. Additionally, potassium enhances disease resistance, stalk strength, and drought tolerance. Optimal soil temperature (50-75°F) fosters bioactivity and nutrient uptake, while a pH range of 5.5–6.5 ensures nutrient availability. Adequate rainfall and irrigation timing influence crop growth, germination, and maturation, contributing to balanced plant development.

3.2 Implementation

In this study, major code and one of the datasets utilized to carry out this study was retrieved from GitHub (B. (n.d.). GitHub). The soil's fertility for crops increases with higher vegetation cover with the unit of percentage (%).
To perform the study, the dataset needs to be filtered and cleaned with the aid of importing Pandas library. The column after 'Vegetation Cover' is considered for soil fertility classification.

```
#renaming column names and units of each attributes
column_names = ['Sample','DIR','INT/EXT','Sub-Sample #','Date','Time','Latitude',
                'Longitude','Slope','Aspect','Vegetation Cover','NO3','NH4','P',
                'K','SO4','B','Organic Matter','pH','Zn','Cu','Fe','Ca','Mg','Na']

#creating new dataframe with column name
soil_data = pd.DataFrame(soil_data.values,columns=column_names)
soil_data.columns
```

Fig. 1: Column names of soil elements

```
#fill null value with 0
soil_data.isna().sum()

soil_data.fillna(0)
soil_data.dtypes
```

Fig. 2: Data cleaning of data set

The dataset is first clean and pre-processed for further analysis. The NULL value is replaced by median value to ease machine learning implementation process. Furthermore, the data is then normalized with Min-Max functions to prevent features with larger magnitudes from dominating the model's learning process. The cleaned, organized data is then stored in a new data frame for further calculation by importing Panda's library.

To further refine the dataset, we imported some functions and classes from the NumPy and sklearn. impute libraries. The columns are input using the median method, and it is shown the 0 missing values remain after this process is completed. This helps to preserve te central tendency of the data distribution thus results in less deviation from the classification accuracy.

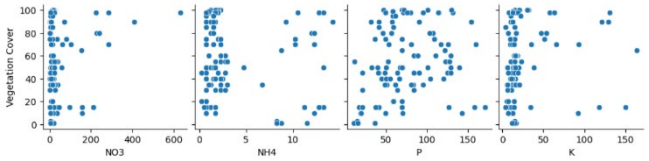


Fig. 3: Pair plots of 'Vegetation Cover' and column 12-15

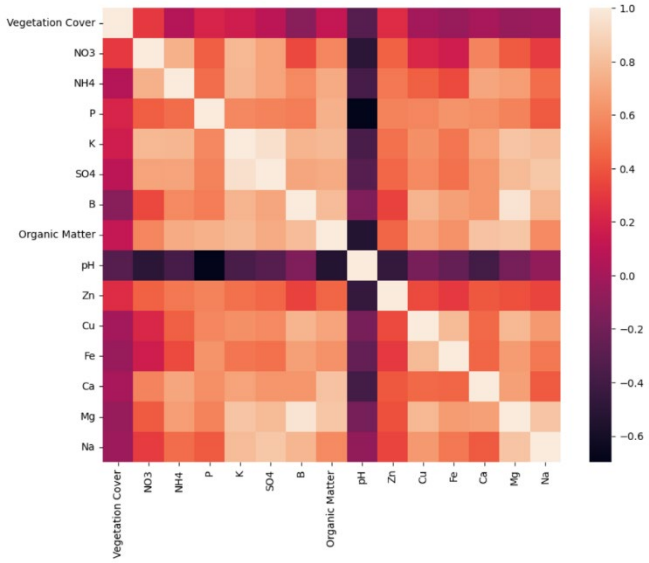


Fig 4: Heatmap represents the correlation between each soil nutrient and vegetation cover.

Additionally, 'seaborn' is imported as 'sns' to aid visualization of soil nutrients weightage among vegetation cover. Firstly, pairplot is utilised to show the relationship between 'Vegetation Cover' and other variables. Second, heatmap is bring to table to visualize the co-relation between each soil nutrient and vegetation cover. The factors influencing vegetation cover are represented in data visualization plot and heatmap reveals the trend and patterns of complex data.

```
import pandas as pd
import numpy as np

X, Y = soil_data[soil_data.columns[11:]], soil_data['Vegetation Cover']

print(X[:10])
print(X)
# Normalizing data
from sklearn.preprocessing import MinMaxScaler
scaler = MinMaxScaler()
X, Y = scaler.fit_transform(X.values),
       scaler.fit_transform(Y.values.reshape(-1,1))
print(X[0])
```

Fig. 5: Data normalization

The needed data is extracted from the original data frame with Pandas library, while MinMaxScaler fuction which performs data normalization imported from Numpy. These 2 are helpfuls tool for data analysis.

Afterwards, Machine Learning and deep learning methods will be implemented as compares for of the results output. Before that, several functions must be imported from sk.learn library to perform machine learning tasks, the code is shown as below:

```
from sklearn.model_selection import train_test_split

X_train,X_test,Y_train,Y_test=train_test_split(X,Y, test_size=0.10, random_state=43)
print(X_train)
```

Fig. 6: Splitting data into training and test set

Splitting data into training and testing sets allows us to train on a subset while retaining another subset to evaluate its performance. Additionally, separation of the data avoids overfit which the model memorizing the training data instead of capturing underlying patterns. Increase the robustness and reliability of the ML training outcomes.

```
from sklearn.svm import SVR

# linear Model
svr_linear = SVR(kernel='linear', C=10, epsilon=0.4)

# RBF kernel
svr_rbf = SVR(kernel='rbf', C=10, gamma=1)

# non-linear model
svr_poly = SVR(kernel='poly', degree=2, C=650, epsilon=0.3)

print('Kernel : Linear')
svr_linear = train(svr_linear,X_train,Y_train)
print_metrics(svr_linear, X_test, Y_test)

with open('SVR (Linear)','wb') as f:
  pickle.dump(svr_linear,f)

print('\n')
print('Kernel : RBF')
svr_rbf = train(svr_rbf,X_train,Y_train)
print_metrics(svr_rbf, X_test, Y_test)

with open('SVR (RBF)','wb') as f:
  pickle.dump(svr_rbf,f)

print("\n")
print('Kernel : Poly')
svr_poly = train(svr_poly,X_train,Y_train)
print_metrics(svr_poly, X_test, Y_test)

with open('SVR (Poly)','wb') as f:
  pickle.dump(svr_poly,f)
```

Fig. 7: SVR ML Method

```
from sklearn.neural_network import MLPRegressor
from sklearn.metrics import mean_squared_error, mean_absolute_error, r2_score

# Create an instance of the MLPRegressor (ANN)
annModel = MLPRegressor(hidden_layer_sizes=(100, 50), max_iter=500)

# Train the model
annModel.fit(X_train, Y_train)

# Predict on the test set
y_pred_ann = annModel.predict(X_test)

# Print metrics
print('ANN Metrics:')
print('Mean Square Error:', mean_squared_error(Y_test, y_pred_ann))
print('Root Mean Square Error:', np.sqrt(mean_squared_error(Y_test, y_pred_ann)))
print('Mean Absolute Error:', mean_absolute_error(Y_test, y_pred_ann))
print('R2 Score:', r2_score(Y_test, y_pred_ann))
print('Accuracy:', annModel.score(X_test, Y_test))
```

Fig. 8: ML Parameters

Specific parameters are considered such as MSE, RMSE, MAE, R2 score and Accuracy. These 5 parameters evaluated the average magnitude of error, variance between dependent and independent variables etc. These parameters provide dissimilar perspectives in evaluating model's effectiveness.

In this experiment, various supervised machine learning method and 2 deep learning method are implemented in soil nutrient classification to determine which mehtods has the best output performance. Machine learning method including Linear Regression, Support Vector Regression, Support Vector Regression (linear, RBF, Poly), Decision Tree Regressor, Random Forest Regression and MLP regressor, while deep learning method includes LSTM (long term short memory) and Simple RNN.

A notable note is that LSTM and Simple RNN are best fit for complex data and sequential pattern over time, these two methods are considered more suitable for future implementations combining real-time data/ remote sensors, preforming classifications in situ. Thus, these 2 DL methods is included only for compares with ML and for future enhancements.

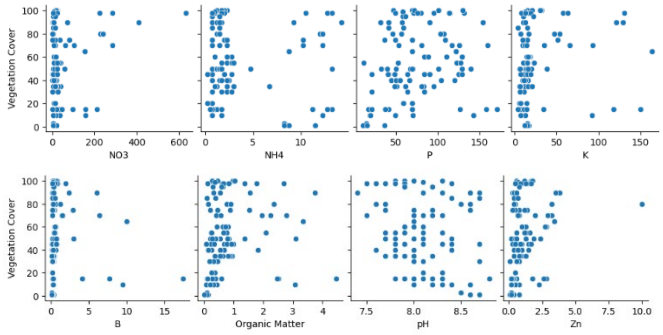
Fig. 9: Match NO3 with N upper and lower limit

And that's not all, we imported another dataset from Crop Recommendation System using LightGBM | Kaggle. for further crop recommendation This dataset contain the major N, P,K, ph value, moisture and temperature values for 22 types of crops. We will need to import 'Latitude' and 'Longitude' values from previous dataset as location reference, and matching the 'NO3', 'P','K' value with the range of upper limit and lower limit (range) of the 'N','P','K' value. The reason behind this is that both datasets extracted from 2 different sources may not have the same but identical parameters. Soil not only contains pure nitrogen but results in the soil nutrient sensor also obtaining mix elements. Both datasets are then merged into a new data frame to obtain crop recommendations data.

3.3 Future Improvement

The agricultural sector can further improve nutrition through several channels: elevating incomes for farming households, expanding crop variety, empowering women who may have less access to traditional resources, and bolstering agricultural diversity and productivity. Additionally, leveraging AI technologies such as ML, IoT, and DL can significantly alleviate workload burdens and bridge knowledge gaps, benefiting women and enhancing overall efficiency. As the AI industry increases efficiency, agriculture sector can transcend traditional gender roles, offering increased employment opportunities for all genders by not just depending on masculine power.

3.4 Results



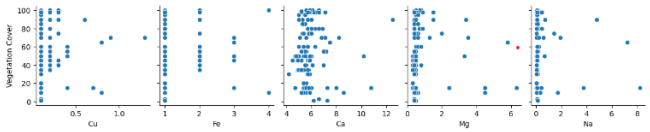


Fig. 10: Pair plot of soil element in vegetation cover

In Figure 10 shows the weightage of soil elements in the given vegetation cover, as soil nutrients does not only consist of N, P, and K value, but also other nano soil elements that can contribute to the crop growing process and boost the harvest process. Hence, all these elements in the figure contribute to comprehensive performance of the crops.

Model	MSE	RMSE	MAE	R2 Score	Accuracy
Linear Regression	0.07	0.27	0.24	0.12	0.12
RF Regression	6.11	2.47	12.95	0.93	0.93
SVR (RBF Kernel)	5.04	2.24	11.89	0.65	0.65
SVR (Linear Kernel)	3.91	1.98	10.81	0.15	0.15
SVR (Poly Kernel)	5.83	2.41	12.54	0.23	0.23
Decision Tree Regression	6.70	2.59	13.54	0.70	0.70
MLP Regressor	0.02	0.15	0.11	0.74	0.74
LSTM	0.12	N/A	N/A	N/A	N/A
Simple RNN	0.10	N/A	N/A	N/A	N/A

Table 1: Results for each model used

The mean square error (MSE), root mean square error (RMSE), and mean absolute error (MAE) show the error value. Hence, the higher the error value, the less efficient the model is, but it is also significant to consider the accuracy value as both are important metrics to justify the performance of the model.

Random forest has the highest accuracy, but also obtains high error value which makes it not compatible as the most efficient model for this system.

Consequently, MLP Regressor is the most efficient model as it has the lowest error value and high in accuracy, which makes it stand out from the performance.

However, comparing deep learning models like LSTM and Simple RNN with traditional regression models can be challenging, and the choice may depend on factors such as the nature of your data and the problem you are trying to solve. Deep learning models may perform better on complex, sequential data, but they also require more data and computational resources which makes it not a suitable technique to be used for this system.

In summarization, for regression problems like soil mapping, lower values of MSE, RMSE, and MAE are desirable, and a higher R2 Score indicates a better fit of the model to the data.

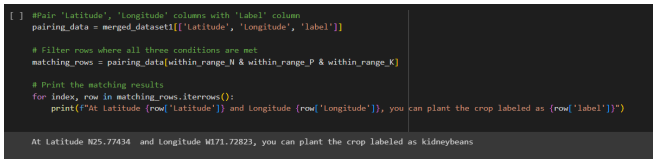


Fig. 11: Result of Crop Recommendation in particular region

All things considered, the specific crop will be recommended to be planted in that area by comparing the soil data extracted from a region (latitude and longitude) match with N, P, K value of the target plants. This approach assists in precise farming, reducing humanized analysis process and farmers' workload. Moreover, implementing AI in agriculture soil nutrient classification greatly reduce the experimental costing

error, which farmers can obtain analysed crop nutrient data, easing the effort to learn new sector of knowledge without becoming the expert in agriculture domain.

4.0 CONCLUSION

In conclusion, the application of Artificial Intelligence techniques which are Machine Learning and Deep Learning applied in the system has proved to be essential in predicting soil fertility and recommending crops based on soil nutrient content. The evaluation of various Artificial Intelligence models revealed their respective strengths and weaknesses, concluding the most suitable model and best performance which is MLP Regressor. Chemical elements such as nitrogen (N), phosphorus (P), and potassium (K), play vital role in soil to ensure optimal growth of plants.

As agriculture continues to face challenges in terms of sustainability and productivity, the integration of Artificial Intelligence technologies offers a promising view in crop yield enhancement. However, further refinement is required to implement the system in real-word agriculture practices. The advancement of Artificial Intelligence helps in promoting sustainability in an efficient way, also contributes in reducing time, cost, and manpower.

REFERENCES

Gupta, A., et al. (2020). Deep Learning Based Crop Recommendation System for Precision Agriculture. Computers and Electronics in Agriculture, 177.

Li, Y., et al. (2021). Spatiotemporal Soil Moisture Prediction Based on Recurrent Neural Networks. IEEE Transactions on Geoscience and Remote Sensing.

Patel, K. B., et al. (2019). A Hybrid Recommendation System for Crop Selection Using Machine Learning and Deep Learning. International Journal of Advanced Computer Science and Applications, 10(8).

Wang, J., et al. (2020). Predicting Soil Nitrogen Levels Using Gradient Boosting Regression: A Case Study in Precision Agriculture. Journal of Plant Nutrition, 43(16).

Zhang, Y., & Wang, L. (2019). Convolutional Neural Network-Based Soil Texture Classification Using Spectral Reflectance Data. Remote Sensing, 11(16).

Zhang, Z., et al. (2018). Predicting Soil Organic Carbon Content Using Machine Learning Algorithms: A Case Study in Precision Agriculture. Journal of Environmental Management, 217.

Niveditha, J. V., Zainudheen, S., Pramodh, S. P., Raneesh, K. Y., Sivan, V., & Hemalatha, S. (2021). Development and testing of soil NPK, moisture and temperature sensing gadget. Journal of Applied Technology and Innovation, 5(3)Top of Form

Njoroge, B. M., Thang, K., & Thiruchelvam, V. (2018). A Research Review of Precision Farming Techniques and Technology. Journal of Applied Technology and Innovation, 2600-7304.

B. (n.d.). GitHub - bhargavi582/-Soil-Fertility-Prediction-Using-Machine-Learning. GitHub. https://github.com/bhargavi582/-Soil-Fertility-Prediction-Using-Machine-Learning .