National College of Ireland

A Sparse-Attention Framework for High-Dimensional Environmental Cost Prediction

MSc Research Project
Msc in Data Analytics

Tejaswini Matta
Student ID: 23395494

School of Computing
National College of Ireland

Supervisor: Dr. Furqan Rustam

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name:	Tejaswini Matta
Student ID:	23395494
Programme:	Msc in Data Analytics **Year:** 2025
Module:	Research and practicum
Lecturer:	Dr. Furqan Rustam
Submission Due Date:	28-01-2026
Project Title:	A Sparse-Attention Framework for High-Dimensional Environmental Cost Prediction
Word Count:	773 **Page Count:** 6

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.
ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature: Tejaswini Matta

Date: 28-01-2026

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST

Attach a completed copy of this sheet to each project (including multiple copies)	□
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	□
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	□

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

A Sparse-Attention Framework for High-Dimensional Environmental Cost Prediction

Tejaswini Matta
Student ID:23395494

1. Introduction

This project develops a regression-based machine learning pipeline to predict environmental cost from a high-dimensional environmental dataset containing operational, ecological, and intensity-related indicators. The task is challenging because the data are typically noisy, partially missing, and heterogeneous, with mixed scales and categories that require careful preprocessing to avoid biased learning. The implemented system performs end-to-end processing cleaning, encoding, scaling, training multiple models from classical baselines to deep learning and attention-based tabular architectures, and evaluating them with standard regression metrics to identify the most accurate and reproducible approach for environmental cost estimation.

2. System Specifications

Hardware requirements

- **CPU:** Multi-core x86_64 processor
- **GPU:** Optional for faster training; recommended NVIDIA GPU with 8 GB VRAM for Tab Net / Transformer / TensorFlow models.
- **RAM:** Minimum 8 GB; recommended 16 GB for smooth experimentation.
- **Storage:** At least 5 GB free space.

Software requirements

- **Operating System:** Windows 10/11, Ubuntu 20.04+/22.04+, or macOS.
- **Python:** Recommended Python **3.10+**.
- **Environment:** Conda or venv; reproducible dependency locking via requirements.txt is recommended.
- **CUDA/cuDNN (only if using GPU):** Install a CUDA/cuDNN combination compatible with your PyTorch and TensorFlow versions.

3. Required Libraries

The pipeline relies on a standard scientific stack for data handling, classical modelling, deep learning, visualization, and tabular attention models. Core libraries include NumPy and Pandas for numerical operations and tabular manipulation; Scikit-learn for preprocessing, train–test splitting, and baseline ML models; TensorFlow/Keras for the DNN, GRU, and LSTM implementations; PyTorch to implement the Tab Transformer and support the Tab Net

backend; and pytorch-tabnet to enable sparse-attention Tab Net regression on structured tabular data. Visualization uses Matplotlib.

The necessary libraries can be installed using the following commands:

a) **pip install numpy==2.3.5 pandas==2.3.3 scipy==1.16.3 matplotlib==3.10.7 seaborn==0.13.2**
b) **pip install scikit-learn==1.7.2**
c) **pip install tensorflow==2.20.0**
d) **pip install torch==2.9.1**
e) **pip install pytorch-tabnet==4.1.0**

4. Dataset description

The dataset comprises firm-level environmental performance indicators for companies in the "Activities auxiliary to financial intermediation" industry category across multiple countries for 2018. It contains quantitative variables representing environmental intensity (relative to revenue and operating income), total environmental cost, and sector-linked production capacity measures (working, fish, crop, meat, wood, and water production capacities). Ecological pressure indicators also form part of the dataset, which are loss of biodiversity, depletion of abiotic resources and imputed percentage estimates which reflect uncertainty in reported values. By and large, it is high-dimensional, heterogeneous tabular data with mixed scales, negative values, and missing/imputed entries features that best suit Sparse-Attention Tab Net that is able to suppress noise and learn powerful interactions among features across various sustainability metrics.

5. Models Implemented

The modelling process has a clear path of development of classical baselines to deep-learning regressors and, lastly, hybrid attention-based tabular models to maintain fair benchmarking and quantifiable gains. The classical machine learning models first to be trained include Decision Tree, Random Forest, Lasso Regression, and Support Vector Regression (SVR) which are models designed to predict on structured tabular inputs with interpretable base performance. Then, deep-learning models such as MLP, a more profound DNN, and gated architectures (GRU and LSTM) are developed to represent the non-linear relationship of higher order after standardized preprocessing. Lastly, explicit models of feature interaction through attention mechanisms Tab Transformer and Sparse-Attention Tab Net are used to explicitly model the feature interaction with Tab Net being able to select features dynamically and generalize well on heterogeneous, mixed-scale sustainability indicators.

6. Code snippets and plots:

1) Data info

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 13177 entries, 0 to 13176
Data columns (total 16 columns):
 #   Column                                                Non-Null Count  Dtype
---  ------                                                --------------  -----
 0   Year                                                  13177 non-null  int64
 1   Company Name                                          13177 non-null  object
 2   Country                                               13177 non-null  object
 3   Industry (Exiobase)                                   13177 non-null  object
 4   Total Environmental Intensity (Revenue)               13177 non-null  float64
 5   Total Environmental Intensity (Operating Income)      12421 non-null  float64
 6   Total Environmental Cost                              13177 non-null  float64
 7   Working Capacity                                      13177 non-null  float64
 8   Fish Production Capacity                              13177 non-null  float64
 9   Crop Production Capacity                              13177 non-null  float64
 10  Meat Production Capacity                              13177 non-null  float64
 11  Biodiversity                                          13177 non-null  float64
 12  Abiotic Resources                                     13028 non-null  float64
 13  Water production capacity (Drinking water & Irrigation Water)  13177 non-null  float64
 14  Wood Production Capacity                              13139 non-null  float64
 15  % Imputed                                             13177 non-null  float64
dtypes: float64(12), int64(1), object(3)
```

2) Source distribution

```
# Count duplicates
dup_count = df.duplicated().sum()
print(f"Number of duplicate rows : {dup_count}")
```

```
Number of duplicate rows : 0
```

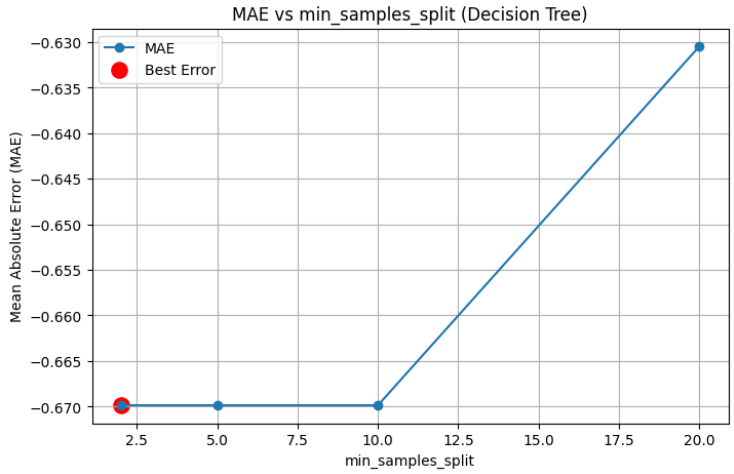
3) Label Encode

```
from sklearn.preprocessing import LabelEncoder

# Columns to encode
cat_cols = ['Company Name', 'Country', 'Industry (Exiobase)']

encoders = {}

for col in cat_cols:
    le = LabelEncoder()
    df[col] = le.fit_transform(df[col])
    encoders[col] = le


df["Total Environmental Cost"] = (df["Total Environmental Cost"].abs() / 1e8).round(2)
# Check dataset
df.head()
```

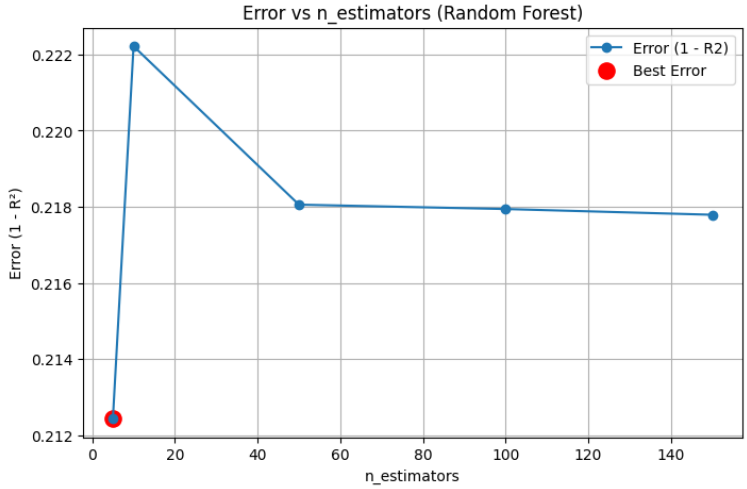
4) Splitting data

```
X = df.drop(columns=['Total Environmental Cost'])
y = df['Total Environmental Cost']

# Train-test split
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

print(" Train/Test Split Complete")
print("Train shape:", X_train.shape)
print("Test shape:", X_test.shape)
```

5) Decision Tree hyperparameter plot



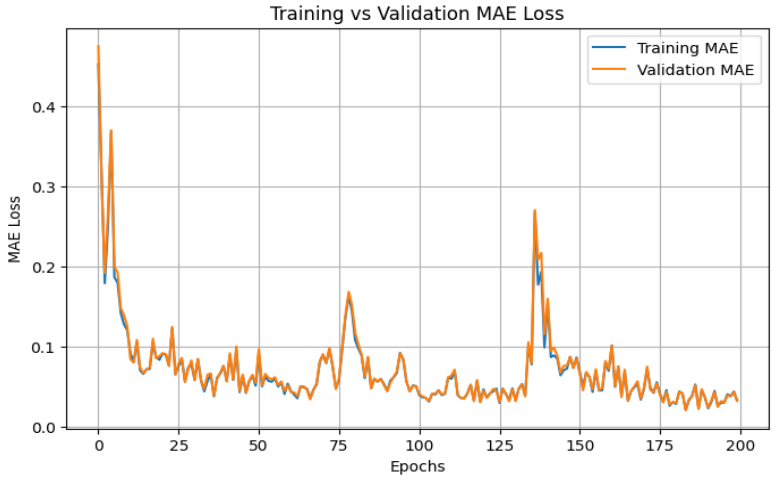
6) Random Forest hyperparameter plot



7) Tab Net Model

```python
import torch
from pytorch_tabnet.tab_model import TabNetRegressor
from sklearn.metrics import mean_absolute_error, mean_squared_error, r2_score

# Convert scaled arrays for TabNet
X_train_tab = X_train_scaled.astype('float32')
X_test_tab = X_test_scaled.astype('float32')
y_train_tab = y_train_scaled.astype('float32')
y_test_tab = y_test_scaled.astype('float32')

# Initialize TabNet
tabnet_model = TabNetRegressor(
    n_d=16,
    n_a=16,
    n_steps=5,
    gamma=1.3,
    lambda_sparse=1e-4,
    optimizer_fn=torch.optim.Adam,
    optimizer_params=dict(lr=2e-2),
    mask_type='sparsemax',
    verbose=1)

```

8) Loss Plot Tan Net Model



9) Comparison of evaluation metrics

Table 1. Performance Comparison of ML Models

Model	R2 Score	MSE	RMSE
Decision Tree	0.82	2432.82	49.32
Random Forest	0.88	1623.38	40.29
Lasso	0.84	2078.72	45.59
SVR	0.79	2793.61	52.85

Table 2. Performance Comparison of DL Models

Model	R2 Score	MSE	RMSE
MLP	0.91	1117.20	33.42
DNN	0.92	999.12	31.60
GRU	0.95	638.16	25.26
LSTM	0.95	609.60	24.69
Tab Transformer	0.96	435.30	20.86
Tab Net Model	0.98	149.59	12.23

7. Conclusion:

This paper presents a completely reproducible pipeline of environmental cost prediction, including preprocessing, model training and standardized evaluation of classical ML, deep learning and attention-based tabular models. Of all the models, the overall performance of Sparse-Attention Tab Net was the best (R 2 = 0.98 with minimal prediction error), which implies that the model can be used to deal with high-dimensional, mixed-scale sustainability indicators due to attention-based feature selection. The workflow can be extended to future datasets and re-used by adding other indicators, alternative validation strategies or explainability modules without altering the basic pipeline structure.

References:

1. https://pandas.pydata.org/docs/
2. https://numpy.org/doc/
3. https://scikit-learn.org/stable/
4. https://www.tensorflow.org/api_docs/python/tf/keras
5. https://docs.pytorch.org/docs/stable/index.html
6. https://dreamquark-ai.github.io/tabnet/