Deep Learning Project for MSc
Molecular Graph Classification

IOANNIS SAVVAS

AM: mtn2319

Abstract

Classifying and predicting the properties of molecular structures using traditional lab experiments is a very costly procedure in many aspects including time. For that reason, computational techniques such as virtual screening have become a go-to method to identify the desired molecules without the need of excessive lab experimentation. Traditional Machine Learning methods are a proven approach to classify molecular structures. However, Graph Neural Networks (GNNs) have emerged as a very promising approach for this task, mainly because molecules can be represented as graph objects with the atoms representing the nodes of the graph and the edges representing the connection between the atoms (bonds). The aim of this project is to create GNNs for binary classification of molecules into readily (RB) and non-readily biodegradable (NRB). A substance is considered readily biodegradable if it can be biodegraded by more than 60% in a 28-day window period. Products that are RB are considered environmentally friendly and preferable over NRB products.

Introduction

Biodegradation is an important factor to take into consideration when designing environmentally friendly products. Approximately 60% of products produced in high quantity contain information about biodegradability [1]. Thus, it's important to develop models to screen chemical substances and have an indication on whether they are RB. A recent study [2] has shown that RB factor can be modelled with high accuracy using standard ML algorithms (kNN, SVM, …) as well as Graph Convolutional Network (GCN). The aim of this project is to develop predictive models using modern GNNs like Graph Attention Network, and FP-GNN and test their performance compared to baseline machine learning models.

Dataset

Overview

The dataset used for the deep learning modelling is comprised of 2 individual datasets merged. The first one [3] is a very popular one that has already been used for modelling and contains 3192

molecules in which 1133 are categorized as RB and 2059 as NRB. The second [4] and most recent dataset contains 3703 structures in which the RB-NRB ratio is 1918-1789. Since the first dataset is quite imbalanced, the second one adds more balance to the total dataset.

Data Split

The NRB-RB ratio of the final dataset is 55-45. To train and evaluate the predictivity of the models that are created it's essential to divide the dataset into training, validation and test in a stratifying way. Thus, the classes ratio percentage is equal in all three of the datasets. The splitting is performed randomly stratified into 80-10-10 sets. The training and validation datasets are used to train and tune the model while the test set is remained hidden for the final evaluation.

Methods

Featurization

To use GNNs, molecules must be encoded into graph objects. More specifically, that is to create the adjacency matrix and the node features-signal for the computation and processing of molecular graphs. For that reason, the RdKIT library was used to create the molecular graphs. Each atom-node contains 68 features that are either one-hot encoded, binary or integers. The features selected were all obtained based on the training set only. The total description of them can be seen in Table 1. Edge features were not used in the models developed.

Atom Feature - Signal	Vector dimension
Symbol	43 (One-Hot)
Adjacent Hydrogens	5 (One-Hot)
Degree	7 (One-Hot)
Formal Charge	1 (Integer)
Radical Electrons	5 (One-Hot)
Hybridization	6 (One-Hot)
Aromaticity	1 (Binary)

Table 1. Node Features

Baselines

Since there are no modelling results available for this custom dataset, baseline models were created using Machine Learning Algorithms. A logistic regression as well as Support Vector Machines were used to fit the data. The features used for this task are molecular fingerprints. Particularly, MACCS fingerprints [5] were used that consist of 167 feature vectors of information for each structure. MACCS were selected because of their low dimensionality to prevent overfitting. They have also been used for the RB modelling before, with good results [3].

Graph Attention Network

The first architecture that was tested is based on the Graph Attention Networks (GAT) [6]. GATs are a special case of Message Passing Networks [7] that utilize self-attention to update node information. Each node is updated according to the equation below:

$$h_i^{(l+1)} = \varphi(h_i^{(l)} \oplus_{j \epsilon N(i)} \; a\left(h_i^{(l)}, h_j^{(l)}\right)\psi\left(h_j^{(l)}\right))$$

where α (self-attention) is the softmax normalized inner product of a learnable attention vector and the concatenation of a linear transformation of the hidden features between two nodes. The attention mechanism allows the model to focus on the most relevant parts of the input and helps the model to weigh the importance of different neighboring nodes when aggregating information. This architecture was chosen primarily because of the high node feature dimension size (multiple different atoms on the training set) so that it can focus on the most important features of the molecule.

FP-GNN

Since molecular fingerprints are based on sets of predefined substructures, they might capture specific patterns that GNNs cannot directly. For that reason, FP-GNN [8] architecture was implemented. It is a hybrid approach that combines a GNNs final representation with a Fingerprint (FP) Based MLP representation to make the final prediction. Graph Attention is used as the GNN module while MACCS fingerprints are inputs to the FP network. The final representation of those 2 architectures is concatenated and passed though a fully connected layer to obtain the logit representation.

Pooling

After the final GNN architecture layer, mean-pooling is applied as a readout function, to generate a coarser graph that can be reduced though a fully connected layer into a single logit. The choice of mean-pooling instead of sum or max pooling was chosen based on experimentation.

Training and Hyperparameter-tuning

Each Deep Learning model was trained on 200 epochs with the use of Learning Rate scheduler to reduce the weight update rate every 20 epochs. Binary Cross Entropy was the loss function selected for the given task optimization. The optimization alogithm used is AdamW with a weight decay value of 0.001. Also, due to the imbalance of the dataset, the loss function was weighted. The model was not selected on the last epoch but rather on the epoch with the best validation loss score.

The hyperparameters of the machine learning baselines were tuned by implementing grid-search. For the Logistic Regression, the C parameter was tuned while for the SVM, the C and the kernel were used. For the Graph Neural Networks, a variety of hyperparameters was chosen to optimize the models. Because of their high training time and computational resources needed for the task,

greedy – search was implemented. The models were trained and evaluated using T4 GPU provided by Google Collab. The hyperparameter range is shown at Table 2.

Name	Range
Batch Size	{32, 64, 128}
Learning rate	{0.0001, 0.0005, 0.001}
Hidden neurons	{16, 32, 64}
Number of attention heads	{2, 3, 4, 5, 6}
Graph Layers	{2, 3, 4}
Activation function	{ReLU, ELU)
Dropout probability	{0.1, 0.2, 0.3, 0.4}

Table 2. Hyperparameters

Results

Machine Learning Benchmarks

After grid-searching the best hyperparameters using the validation set, the best machine learning models were created to be evaluated on the unseen test set. For the Logistic Regression model, the balanced accuracy was 0.809 while for the SVM model 0.828. By observing the confusion matrices in figures 1,2, it is clear both models accuracy on the RB class is the same. However, SVM was superior when predicting the NRB class.

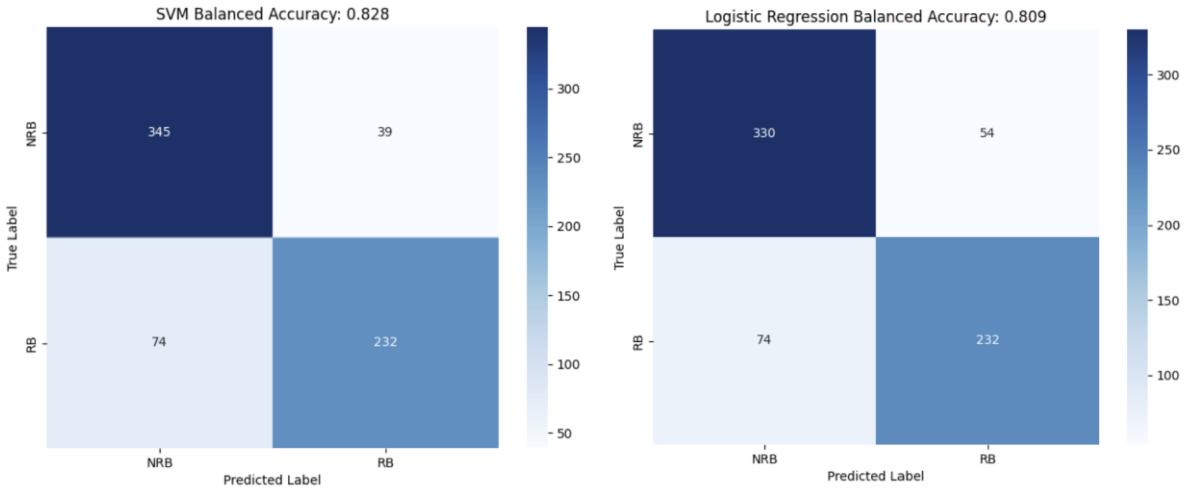

Figure 1,2: Left: SVM confusion matrix, Right: Logistic Regession confusion matrix

Graph Attention Network

Seeing Figure 3, GAT appears to be learning thoughout 200 epochs. However, in Figure 4, NRB is overpredicted and the model fails to capture information about NRB molecules as good as simple Machine Learning models.

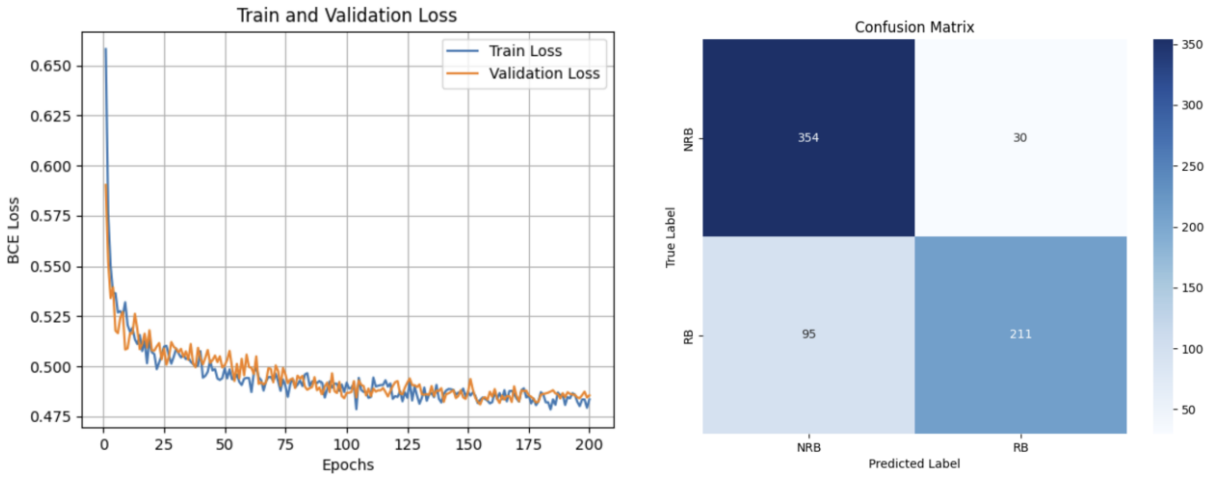

Figure 3,4: Left: GAT Loss Curve (Training, Validation), Right: Confusion Matrix of the test set

FP-GNN

It is obvious that FP-GNN learning (Figure 5) is much smoother than GAT where both the training and validation curves are dropping simultaneously. Also loss values are much lower than GAT. After training, this is also printed in the confusion matrix (Figure 6) where both True Positives and True Negatives are better than standard ML while overpredicting is not occurring compared to GAT.

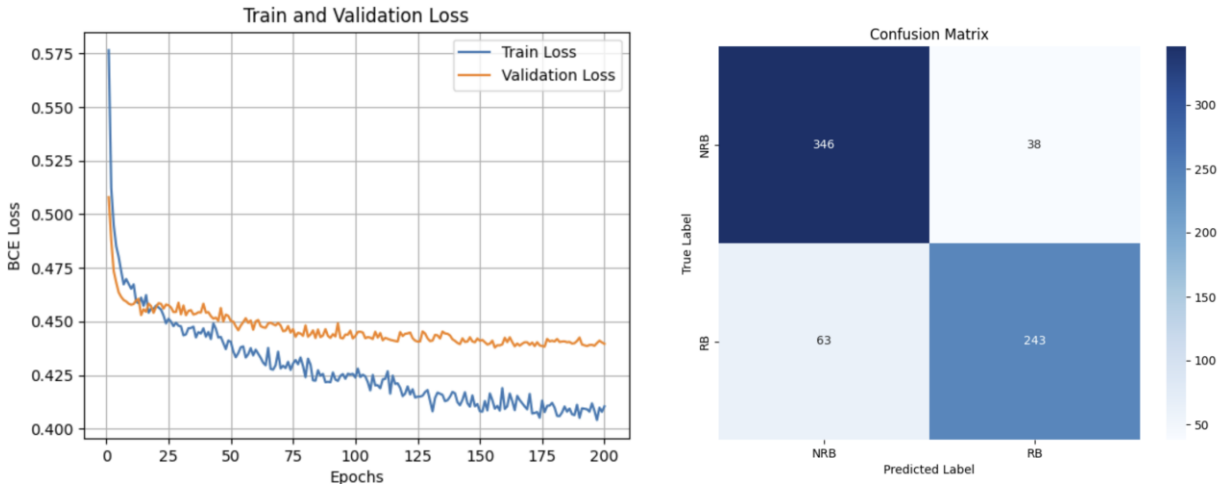

Figure 5,6. Left: FP-GNN Loss Curves (Training, Validation), Right: Confusion Matrix of the test set

Comparison and conclusion

Model	BA	Validation loss	Training loss
LR	0.809	-	-
SVM	0.828	-	-
GAT	0.806	0.481	0.483
FP-GNN	**0.848**	**0.438**	**0.409**

After creating 4 individuals models, it is clear that FP-GNN outperforms all of them. GAT network doesn't seem to have any superiority than common machine learning algorithms. That is probably due to its inability to capture underlying substructures inside a molecular structure. However, when combining the structural information of the GAT with the MACCS fingerprints the learning improves. Not only that, but the accuracy is better in both classes, especially in the positive class where imbalance lies. It is safe to say that GAT might have failed to capture specific structural motifs or functional groups. FP-GNN architecture can be further utilized by adding more fingerprint bits [8] and not only MACCS. The source code and the results (hyperparameters, diagrams, etc.) are provided in https://github.com/johnsaveus/GNN_biodegradability.

[1] R. ALLANOU, B. G. HANSEN, and D. B. Y. VAN, "Public Availability of Data on EU High production Volume Chemicals - Part 2." Accessed: Jun. 23, 2024. [Online]. Available: https://publications.jrc.ec.europa.eu/repository/handle/JRC27013

[2] M. Lee and K. Min, "A Comparative Study of the Performance for Predicting Biodegradability Classification: The Quantitative Structure–Activity Relationship Model vs the Graph Convolutional

Network," *ACS Omega*, vol. 7, no. 4, pp. 3649–3655, Feb. 2022, doi: 10.1021/ACSOMEGA.1C06274.

[3] "Modelling of ready biodegradability based on combined public and industrial data sources", doi: 10.5281/ZENODO.3540701.

[4] "An Expanded Dataset for Improved Prediction of Chemical Biodegradability", doi: 10.5281/ZENODO.8255910.

[5] J. L. Durant, B. A. Leland, D. R. Henry, and J. G. Nourse, "Reoptimization of MDL keys for use in drug discovery," *J Chem Inf Comput Sci*, vol. 42, no. 6, pp. 1273–1280, Nov. 2002, doi: 10.1021/CI010132R.

[6] S. Brody, U. Alon, and E. Yahav, "How Attentive are Graph Attention Networks?," May 2021, Accessed: May 16, 2023. [Online]. Available: https://arxiv.org/abs/2105.14491v3

[7] J. Gilmer, S. S. Schoenholz, P. F. Riley, O. Vinyals, and G. E. Dahl, "Neural Message Passing for Quantum Chemistry," 2017.

[8] H. Cai, H. Zhang, D. Zhao, J. Wu, and L. Wang, "FP-GNN: a versatile deep learning architecture for enhanced molecular property prediction," *Brief Bioinform*, vol. 23, no. 6, May 2022, doi: 10.1093/bib/bbac408.