

[논문리뷰]TabNet [2019]

Introduction

☀ 해당 논문은 **novel high-performance and interpretable canonical Tabular data**에 관한 **Deep Learning architecture**인 **TabNet**을 소개합니다.

TabNet은 Sequential attention을 사용하여, **각 의사 결정단계에서 해석 가능성 및 기타 기능 지원 학습능력이 가장 많이 사용되는 효율적인 학습**을 진행합니다.

FC(Fully Connected layer)를 기반한 DNN(Deep Neural Network)는 표 형태의 tabular data를 신경쓰지 않습니다.

- Decision tree 기반의 ensemble은 tabular data를 지배하고 있습니다.
 - Tabular data는 decision manifold에서 효율적입니다.
 - Tabular data는 해석(interpret)하기에 매우 쉽습니다.
- DNN의 경우 과도하게 parameter화 되어, best Optimizer를 찾기에 상당히 어렵습니다.

But, Instance(사례)가 많아진다면, 다른 방법들보다 좋은 성능을 나타낼 수 있습니다. 표 형식 데이터를 다른 형식의 데이터와 함께 효율적으로 학습하거나, 다른 목적의 NN 학습 방법을 적용하는 end-to-end learning이 가능합니다.

- TabNet은 전처리 과정이 필요하지 않습니다.
- Decision step을 통하여 feature Selection을 진행합니다.
- 무작위로 feature값을 예측하는 unsupervised pretrain 단계를 적용하여 **상당한 성능 향상**을 보여줍니다.
- **Sequential Attention**을 활용하여, 각 **Decision Step의 Feature Selection 설명**이 가능하고, 더 잘 학습할 수 있다고 합니다. 이러한 **feature selection**은 **instance-wise**하게 수행됩니다.

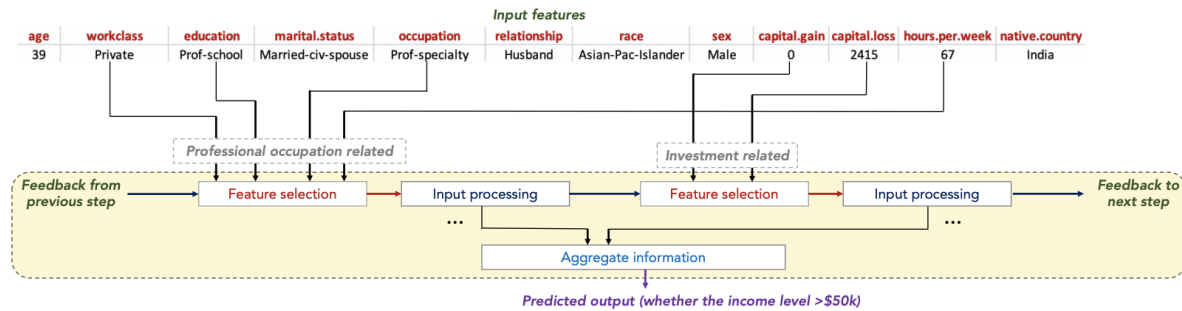


Figure 1: TabNet's sparse feature selection exemplified for Adult Census Income prediction (Dua and Graff 2017). Sparse feature selection enables interpretability and better learning as the capacity is used for the most salient features. TabNet employs multiple decision blocks that focus on processing a subset of input features for reasoning. Two decision blocks shown as examples process features that are related to professional occupation and investments, respectively, in order to predict the income level.

Related Work

Feature selection

- Feature Selection은 predict에 대한 usefulness를 기반으로, 기능의 하위집합을 광범위한 부분에서 신중하게 선택합니다.
- 기존 Feature Selection 방식은 모델에서 분리된 형태를 가지고 있으나, TabNet은 Single Model에서 같이 학습되는 구조를 가지고 있습니다.

Tree-based Learning

- 기존 Tree기반 모델들은 모델의 분산을 줄이면서 성능의 향상을 이끌었습니다. (XGBoost, LGBM)

Integration of DNNs into DTs:

- 빨강 Mask 박스를 보면, 첫 번째 Mask에서는 첫 번째 계수만 1 나머지는 0, 두 번째 Mask에서는 두 번째 계수를 1로 두어 다른 feature의 weight를 제거하여, **Feature Selection으로 결정 경계를 만들어가는 트리의 학습 방법과 비슷하게 만들 수 있다고 합니다.**

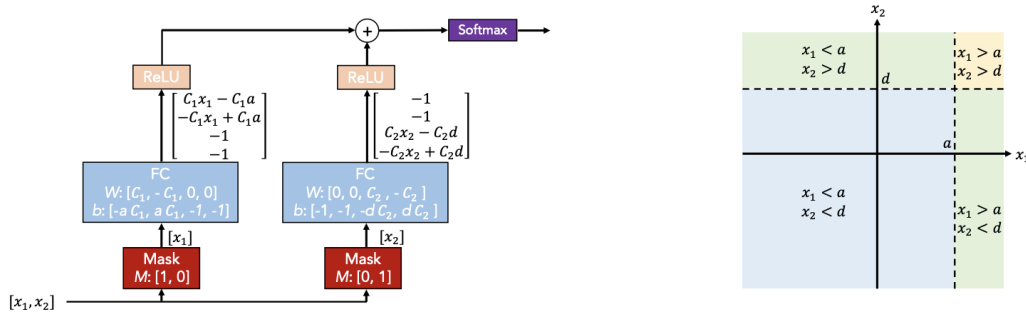


Figure 3: Illustration of DT-like classification using conventional DNN blocks (left) and the corresponding decision manifold (right). Relevant features are selected by using multiplicative sparse masks on inputs. The selected features are linearly transformed, and after a bias addition (to represent boundaries) ReLU performs region selection by zeroing the regions. Aggregation of multiple regions is based on addition. As C_1 and C_2 get larger, the decision boundary gets sharper.

Self-Supervised Learning

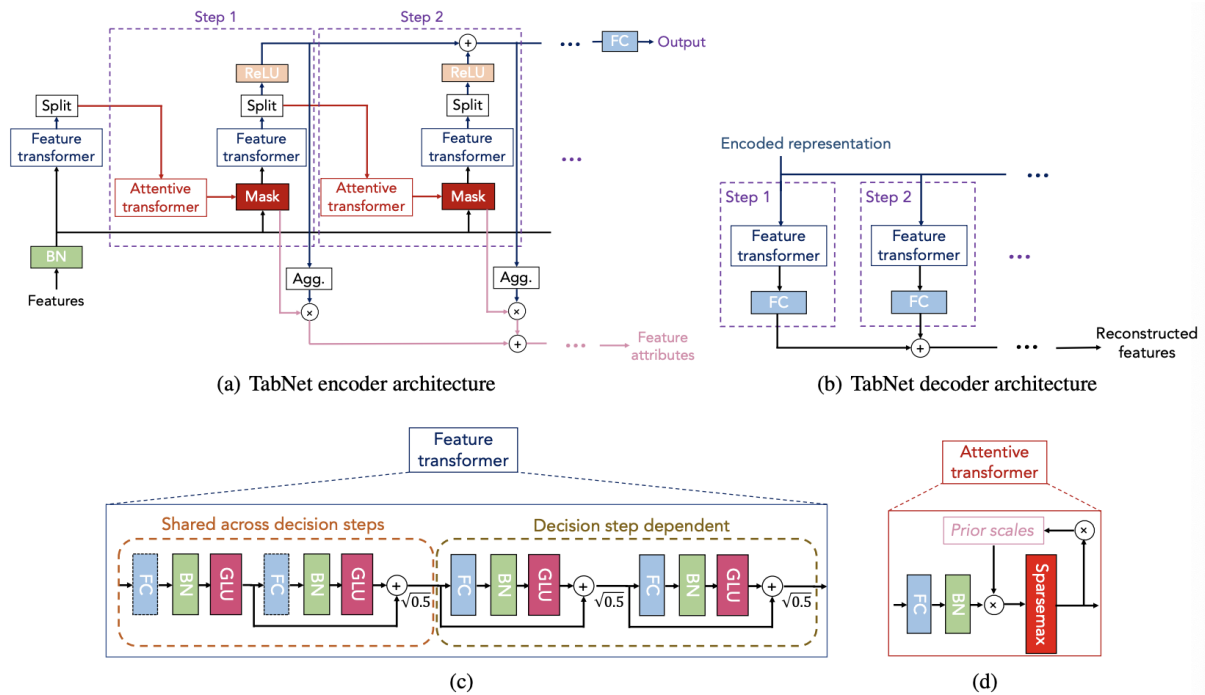
- Unsupervised Learning은 Supervised Learning을 소규모 데이터셋에서 개선하는 것이 가능합니다.

TabNet for Tabular Learning

TabNet의 Encoder부분은 Feature Transformer, Attentive Transformer, Feature Mask로 이루어진

Decision Step으로 구성되어 있습니다.

Decoder는 Feature Transformer로 구성되어 있습니다. FC, BN, GLU를 단위로 구성되며, shared step에서는 모든 Step에서 공유되며, Dependent Step에서는 앞의 layer에 의존합니다.



Advantage

- 데이터에서 학습되는 sparse feature selection을 활용합니다.
- 순차적 다층 아키텍처를 구성해서 정보를 통합합니다.
- 선정된 feature의 비선형 처리를 통해 학습 능력을 향상합니다.
- 범주형 데이터를 입력 할 경우, Embedding Feature를 매핑합니다.

Conclusion


Model	Test accuracy (%)
DT	50.0
MLP	50.0
Deep neural DT	65.1
XGBoost	71.1
LightGBM	70.0
CatBoost	66.6
TabNet	99.2
Rule-based	100.0

Model	Test MSE
MLP	512.62
XGBoost	490.83
LightGBM	504.76
CatBoost	489.75
TabNet	485.12

TabNet은 Sequential attention mechanism을 활용함으로써, 의미론적으로, 의미있는 feature를 선택합니다. 또한 Unsupervised pre-training이 상당한 benefit이 있음을 증명하였습니다.

TabNet: Attentive Interpretable Tabular Learning

We propose a novel high-performance and interpretable canonical deep tabular data learning architecture, TabNet. TabNet uses sequential attention to choose which features to

 <https://arxiv.org/abs/1908.07442>

