Deep learning for tabular data

Deep learning has already obtained tremendous success in various domains including computer vision (CV), natural language processing (NLP), and reinforcement learning (RL). Making use of a large amount of data and modern neural network architectures, it is possible to discover intrinsic dependencies in data and learn a powerful and abstract representation of complex data. Consequently, machine learning models have been reported to exhibit or even surpass human-level performance on individual tasks, such as Atari games or object recognition.

In real-world applications, the most common data type is tabular data, comprising samples (rows) with the same set of features (columns). Tabular data is used in practical applications in many fields, including biology, medicine, finance, manufacturing, and many other applications that are based on relational databases. However, deep learning models are mostly designed for CV and NLP tasks, which represent only a subset of real-life data. According to recent reports, data science and machine learning developers work with tabular data as often as with texts or images[1]. Even though practical applications are dominated by tabular data, deep learning models do not confirm the state-of-the-art performance in this domain. Indeed, traditional ensemble models based on decision trees, like XGBoost, remain the go-to tool for most practitioners[2].

There are many potential reasons why the progress of deep learning in CV and NLP is not reflected in the domain of tabular data. Modern deep learning architectures, such as convolutional networks, recurrent neural networks, or transformers, emerged after years of research to create inductive biases that match invariances and spatial dependencies of image and text data. **Finding corresponding invariances and local dependencies in tabular data is hard**, which makes the fully-connected architectures the first choice for tabular datasets. Moreover, typical deep learning models containing millions of parameters are trained on an enormous amount of data coming from common domains, such as photographs in the case of CV, which allows them to discover sophisticated patterns without overfitting. In real-world settings, **small tabular datasets are ubiquitous**. If the dimension of data is relatively large compared to the number of examples, then neural networks rapidly overfit, which prevents them from using deeper architectures. The existence of **heterogeneous features with extreme values and data sparsity** causes additional difficulties in training deep learning models on tabular data. In consequence, well-established tree-like methods, such as XGBoost or Random Forests are considered the recommended option for real-life tabular data problems.

Motivated by the importance of tabular data in real-life problems, we aim at transferring the successful deep learning story from CV and NLP to the case of tabular data.

In this project, we do not restrict ourselves to any specific application or problem. We will focus on constructing general deep learning models, which can be used in various tabular data problems. In particular, we focus on the following goals:

- Building deep learning models for solving large-scale problems in tabular data.
- Designing deep learning models specialized for small tabular data.
- Constructing weakly-supervised models, which can learn a powerful representation with a minimal amount of information delivered by humans.
- Increasing the interpretability of tabular data.

The created methods and algorithms will be verified and applied to various real-life problems. In particular, we will work on human metagenomic data, which describes the composition of the gut microbiome. We will also consider medical databases containing patients' data and the task is to predict the presence or risk of specific illnesses such as cancer, irritable bowel syndrome, and mental health disorders. Finally, we will work on data generated from sensors monitoring machine behavior. In consequence, the project has an interdisciplinary character, and its outcome will also have an impact outside pure machine learning.

[1] https://www.statista.com/statistics/1241924/worldwide-software-developer-data-uses/

[2] https://www.kaggle.com/kaggle-survey-2021