IGI | Institute of Theoretical Computer Science

Compact prediction learning

Deep neural networks (DNNs) are powerful machine learning models. Representations in hidden layers of DNNs are however strongly distributed, which renders them hard to interpret and reuse. It is believed that more symbolic representations would improve interpretability and potentially also out-of-distribution generalization.

In this project, we will investigate options to learn more symbolic-like state-representations in hidden layers of DNNs.

Consider a set of inputs $X = \langle \boldsymbol{x}^1, \ldots, \boldsymbol{x}^K \rangle$ and corresponding targets $T = \langle t^1, \ldots, t^K \rangle$.

A neural network (NN) computes a function $h(\boldsymbol{x})$ to approximate this function. We split the NN into an encoder $g(\boldsymbol{x})$ which produces a representation of $\boldsymbol{x}$ with some desired properties and a classifier f that takes the representation as input and produces the output. Hence the network computes $f(g(\boldsymbol{x}))$.

We want to find a g that maps X to a minimal set $Y = \{\boldsymbol{y} | \exists \boldsymbol{x} \in X: g(\boldsymbol{x}) = \boldsymbol{y}\}$ while still performing the overall task.

In general, we will minimize the error of the network output (e.g., cross-entropy error) together with an error on the representation that enforces a compact representation of the inputs. One such “representation”-error could be

$$E = \sum_{k \neq l} \log\left(1 + \sum\nolimits_i \left| g_i(\boldsymbol{x}^k) - g_i(\boldsymbol{x}^l) \right|\right),$$

where $g_i(\boldsymbol{x})$ is the i-th component of $g(\boldsymbol{x})$. This loss function approximates a loss that is defined by the size of the set Y given above.

We will apply this principle in the context of predictive learning.

Goals & Tasks

- Implement compact prediction learning in TensorFlow or PyTorch.
- Perform simulations and document results

Qualifications

- Interest in deep learning.
- Experience with Python based deep learning frameworks such as TensorFlow or PyTorch.
- Registered to one of the following:
  - ✓ **Bachelor Thesis**
  - ✓ **Seminar Project**
  - ✓ **Master Thesis**

Contact

Robert Legenstein
Robert.legenstein@igi.tugraz.at