

Unsere Kompetenzfelder

Künstliche Intelligenz und maschinelles Lernen

Software-Entwicklung für künstliche Intelligenz und maschinelles Lernen

Methoden und Tools

- » Python/Java/Javascript
- » Tensorflow/TF Lite/Keras/PyTorch
- » CNN/RNN/LSTM/Transformer Reinforcement Learning
- » Tensorboard/Neptune.ai
- » panas/numpy/Seaborn/ydata profiling
- » matplotlib/plotly
- » Apache Spark/PySpark
- » MLFlow/Weights & Biases
- » Evidently/Fiddler
- » Nvidia Cuda
- » AWS/Google Cloud
- » Jira/Confluence/Bitbucket/Git

Implementierung neuronaler Netze



- Nutzen **gängiger Frameworks** (TensorFlow, Keras, PyTorch)
- **Training und Analyse**
- Große Erfahrung mit **verschiedensten Netzarchitekturen**

Netzwerk- architekturen für Supervised Learning



- **CNNs** für Bild- oder Audiodaten
- **RNNs, LSTMs oder Transformer** für die Sequenzanalyse oder Prädiktion
- **Siamese Triplet Networks** für Unterscheidungsaufgaben
- **Feed Forward Networks** für die Regression/Klassifikation

Reinforcement Learning



- **Aufsetzen und Trainieren** selbstlernender Netze
- **Trainingsanalyse und Optimierung** selbstlernender Netze

Data Science



- Analyse **großer Datenmengen**
- **Datenbereinigung** und **-visualisierung**
- Aufzeigen von **Korrelationen**
- Aufzeigen von **Optimierungspotenzialen** basierend auf Ergebnissen

Verifikation und Analyse trainierter Netze



- **Trainingsoptimierung**
- Überprüfung und Visualisierung der **Modellstabilität** im laufenden Betrieb
- **Interpretationsanalyse** von Ergebnissen