YURII PETROV

CONTACT

Kyiv, Ukraine
380989115943
tonyandrew0308@gmail.com
portfolio-dv-git-main-waritos-projects.vercel.app
Telegram:
https://t.me/devfancyocean0308

Language

English
Ukrainian

EDUCATION

BACHELOR IN COMPUTER SCIENCE, 04/2013 - 06/2018

Odessa Mechnikov National University, Odessa, Ukraine

MASTER'S IN SOFTWARE ENGINEERING, 07/2018 - 05/2020

Institute of Information Sciences and Electronics, Odes'ka, Ukraine

KEY SKILLS

Python, Django, AWS, S3/SDM, Spark/DP, ORM, TensorFlow, PyTorch, scikit-learn/ML, Keras/DL, OpenAI, Gym/RF, RLlib/RF, OpenCV, Google, AI, Platform, AWS, SageMaker, Microsoft, Azure, Machine, Learning, spaCy/NLP, NLTK/NLP, Gensim/NLP, Docker

INTERESTS

Full-Stack Development

PROFILE

Seasoned AI Engineer with over 7 years of experience in ML, FL, DL, and implementing web-based service and solutions. Proficient in both AI Engineer and Full-Stack development. Adept at collaborating with cross-functional teams to deliver high-quality, scalable, and maintainable software solutions. Strong problem-solving skills with a keen eye for detail, ensuring seamless integration and performance optimization. Passionate about learning new technologies and staying updated with technology trends to drive innovation and deliver cutting-edge solutions.

EXPERIENCE

DJANGO STARS • 07/2020 - 04/2024

Kiev, Ukraine

Senior AI Engineer

Crafting an ensemble stack of machine learning models to analyze the Kyiv air pollution dataset. The stack ensemble included a diverse mix of linear models, tree-based models, support vector models and neural networks as base models. The final meta model was the perennial favorite OLS. The target variable was the one-hour ahead PM 2.5 pollution reading. The data exhibited significant outliers and complex seasonality, and was plagued by missing target values. Following pre-modelling data analysis and engineering, the data was subsetted three-ways, according to its temporal order, with the latest 10% of the data taken as the holdout test set. The remaining 90% of the data was in turn split into an earlier gridsearch training set (2/3) for the base models, and a later meta training set (1/3) for the meta model. The training data (the 90%) was also used to run a 5-fold forward chain cross-validation procedure to assess model performance, for all the models used.