



Aliaksandr Hvorashch

Data Scientist
(Junior)

Contacts

Mob:

+375 (29) 567-67-34

Email:

saatarko@tut.by

Telegram:

@Saatarko

LinkedIn:

www.linkedin.com/in/saatarko

GitHub:

https://github.com/Saatarko

About me

I decided to change my profile due to changes in the principles of work at my main place of work..

- Since childhood, in my free time I have been interested in programming, including attending various courses and trying myself in writing mini games (QSP).
- I have now completed courses on Data Scientist. I plan to continue developing in this specialty.
- I am taking courses on:
 - [Stepik](#)
 - Coursera
- I'm studying:
 - RL/CV/NLP
 - competition on [Kaggle](#)
 - articles [geeksforgeeks.org](#)

Education

Data Scientist

Teachmeskills Company
December 2024 - September 2025

Convolutional Neural Networks

Coursera (Andrew Ng)
May 2025 - June 2025

ML

Stepik (Sergey Balakirev)
May 2025

Python Developer

Teachmeskills Company
April 2024 - September 2024

Experience

Data Scientist (graduate)

Teachmeskills Company
December 2024 - September 2025

Skills (ML/DS):

- Good knowledge: classic ML models (Logistic Regression, Random Forest, XGBoost, CatBoost), data processing (Pandas, NumPy, scikit-learn).
- I use the following skills: PyTorch (neural networks, CNN, RNN, transformers), OpenCV, HuggingFace (NLP).
- I have experience working with: recommended systems, Reinforcement Learning (including multi-agent prototypes).

Pet-project: [Recommender system](#)

- Developed a prototype of a recommended system for analyzing user preferences.
- Approaches used: collaborative filtering, content-based recommendations, hybrid methods.
- Technologies: Python, Pandas, scikit-learn, PyTorch.

Diploma (solo):

[RL Multi-Agent Maze Prototype](#)

- Topic: Research of multi-agent architecture with separation of functions (environment generation, problem solving in the environment, meta-control).
- A prototype environment with elements of maze generation and agent training via Reinforcement Learning has been implemented.
- Main framework: PyTorch (Policy Gradient, PPO used).

Result: a working experimental environment for further research of multi-agent systems.

Python Developer (graduate)

Компания Teachmeskills
апрель 2024 г. - сентябрь 2024

Skills:

- Experience with Python Backend (Django/ Flask/FastAPI)
- Good knowledge of DB: SQLite, MySQL, PostgreSQL
- Use Redis and Celery

Results:

- [Diploma \(in teams\):](#)

Social network with profiles, news, groups and group chat on Websocket. The purpose of the diploma is to consolidate and develop knowledge of Django

- Technologies Used:
Django, SQLite, Redis, Celery, HTTP, WebSockets

- Diploma (solo):
IT school. Backend on FastAPI + "frontend" API with courses, course registration, group enrollment and chat on Websocket

The purpose of the diploma is to consolidate and develop knowledge of FastAPI

- Technologies Used::
FastAPI, SQLite, Redis, Celery, WebSockets, Kivy

- A small network game for 2 people - "shootout"

Technologies used:
Python, Pygame, Sockets,

Experience

Director

Geoportal LLC
November 2009 - present

Skills:

- Enterprise Management
- Conducting technical training for customers
- Translation of technical documentation

Results:

- Organized technical support and customer training
- Large number of supported manufacturers

Hard Skills

Programming languages:

Python (Python Core)

Backend development

Django/ Flask/FastAPI

English proficiency level:

B2

Machine Learning / Data Science:

- Working with classic models (Logistic Regression, Random Forest, XGBoost, CatBoost).
- Data processing: Pandas, NumPy, scikit-learn.
- Computer vision (OpenCV, CNN).
- Natural Language Processing (NLP): transformers (BERT, GPT, HuggingFace).
- Recommender systems (collaborative filtering, content-based, hybrid approaches).
- Reinforcement Learning (RL): reinforcement learning, including multi-agent systems.
- Main framework: PyTorch (TensorFlow – basic experience).
- ML ecosystem: Jupyter, Matplotlib, Seaborn.

Additionally::

- Asyncio, multiprocessing, threading
- Protocols: HTTP, WebSockets
- DB: SQLite, MySQL, PostgreSQL/MongoDB
- Caching and tasks: Redis and Celery
- Working with Docker/Docker compose.GitHub
- Kivy, PyGame, PySide6/PyQt

I have an idea about:

- JavaScript (including React.js)
- AWS (Lambda, S3, CloudFront)
- CI/CD
- Kafka / RabbitMQ