

Julian Kozłowski

❖ 575999680 ❖ Konstantinów Łódzki, 95-050 ❖ jkozowski.com ❖

EDUCATION

The University of Warsaw

Year 2019-2022

BSc., Mathematics

- Completed degree within the Interdisciplinary Faculty of Mathematical and Natural Sciences
- Specialized in Probability Theory and machine learning, culminating in a thesis focused on the advancements and evaluation of Reinforcement Learning algorithms.
- I was a member of "Koło pasjonatów matematyki" - in which I helped with organizing "douczki" for younger students to help them with material on upcoming exams - I was a tutor for a probability theory course.

The University of Warsaw

Year 2020-2023

BSc., Informatics

Medical University of Łódź

Year 2023-ongoing

Msc. Medicine

WORK EXPERIENCE

Iterators

Summer 2022

Backend intern

Warsaw

- Assisted in the regular maintenance of the mobile application called "Obi" by diagnosing and resolving software bugs to ensure optimal functionality and user experience.
- Utilized a robust stack that included Scala, internal libraries (like Klibs), Akka, and Cats.

Mim solutions

February – May 2023

Junior Data Scientist

Warsaw

- Contributed to a project aimed at matching tweets with corresponding articles, which culminated in a paper submission. My responsibilities included implementing a LDA baseline model and developing a PyTorch implementation for the primary model. The architecture of our model was similar to Open AI's CLIP architecture.
- Contributed to the flow of other teams by helping with the incorporation of tools such as Hydra, Pytorch Lightning, docker, and Ruff.
- Helped with conceptual work on theoretical foundations of models.

Some projects - all available publicly on my GitHub account: github.com/jkozlovski

- Implementation of a decoder-only transformer model based on sequences generated in a random manner detailed by a Markov chain. Based on <https://arxiv.org/abs/1706.03762>.
- Implementation of backpropagation algorithm inspired by: <https://github.com/cybertronai/gradient-checkpointing>
For the cost of complexity, we reduce memory usage, such as back-propagation algorithms are used when there is a need for a bigger net but there is not enough space to hold the weights in the memory of the GPU.
- A simple integration system is written entirely in Rust. Inspired by: <http://aosabook.org/en/500L/a-continuous-integration-system.html>

Interests

- Psychology and neurosciences, gym rat 🐭, puzzles, math problems not too difficult to give up but not too trivial to be solved under 30 min. mark