+44 7305470094
Leeds, UK
wiciakMichael@gmail.com

Michael Wiciak

Software Engineering

MichaelWiciak.com
Github.com/MichaelWiciak
linkedin.com/in/michael-wiciak1

Proficient in Python (expert), C (proficient), C++ (competent), Java (competent), TypeScript/JavaScript (competent), Bash (competent), and MATLAB (competent); basic knowledge of Go and Rust, applied in various public GitHub projects. Open to remote work. Fluent in Polish and English; basic proficiency in Chinese.

Education

MEng/BSc Computer Science with Artificial Intelligence (09/2021 - 06/2025)
University of Leeds - *First Class / 4.00 GPA (Expected)*

- BSc Dissertation: "Developed and fine-tuned CodeBERT, RoBERTa, and UniXCoder models for code completion using Masked Language Modelling. Optimised fine-tuning speed by 50% with LoRa on NVIDIA P100 GPUs. Designed a reusable pipeline for fine-tuning on new data. Discovered that accuracy decreases after 100 tokens in docstrings, with exponential increases in generation time for longer sequences, except in AST models. Found that code-comment models performed best when predicting multiple tokens consecutively. Achieved 81% accuracy on CodeSearchNet and 70% on CodeNN (C++ not included in training). Designed a general fine-tuning algorithm for multiple Transformer-based models." Achieved a First-Class mark.
- Module grades: 96% Object Oriented Programming, 95% Computer Processors, 86% Artificial Intelligence, 83% Machine Learning, 95% Discrete Mathematics, 84% User Interfaces, 82% Software Engineering Project.

The Grammar School at Leeds (09/2019 - 06/2021)

- *A-levels*: A*A*AA in Computer Science, Mathematics, Economics, and Extended Project Qualification

Work Experience

***Co-Founder* - TimetableX** (10/2022 - Present) *timetablex.com* Demo Video

- Co-founded and launched TimetableX (Business Number: 15186230), attracting 200 users with 50 daily active users. Developed a feature allowing students to share assessment details publicly within modules, making coursework visible to others in the same module, with automated reminders via an internal notification system. Created a persistent to-do list, customizable UI with over 30 themes, adaptable for both desktop and mobile resolutions. Coded in Typescript and React, Python.
- Developed a web scraper to map room locations from iCal data, caching events for 24 hours with background updates using Cron. Implemented a MongoDB model with 11 collections, adhering to 3rd Normal Form, and integrated Prisma ORM. Enabled timezone adaptability based on the user's device. Opened discussions with the University of Leeds to improve their timetabling system, with TimetableX supporting both Leeds and York universities. Integrates auth0 for authentication, stripe for payments, trpc for API. Generates ical links to allow user to migrate to other services while maintaining the event/assessments they added.

Car Engineering Tutor - TechCamp (07/2022 - 08/2024, Summer job)

- Taught a total of 50 under-18 students. Designed and implemented the transmitter and receiver code using ESP-NOW, a connectionless protocol utilising Wi-Fi frequency for peer-to-peer communication on ESP32 microcontrollers. Achieved a reliable communication range of 100 meters. Refactored code to enable students to customise the RC cars' behaviours based on button presses and programming animations on a monochrome display with u8g2. Guided students through soldering and assembling custom Tamiya RC Cars and programming of two PCBs using C and C++.

Projects

Sports Booker – *sportsbooker.timetablex.com* - Demo Video

- Facilitated over 6,000 automated bookings for badminton courts at The Edge Gym in Leeds, serving two private clubs with a total of 50 users. Leveraged Microsoft Azure VM for cron jobs and Docker to initialize the database using PostgreSQL. Developed an API for user data retrieval and bookings, utilizing Next.js and T3 App for a robust front-end experience. Implemented bots using Chromium to navigate the outdated Edge system for mass bookings and cancellations, with a flexible scheduling feature to adjust bot behaviour. Created a Discord bot to notify users of booking failures and integrated a notification system for the website. Additionally, offered a command-line interface (CLI) for direct bookings through the API.

Turtle Bot in an Exploding Ship – Demo Link

- Implemented a robot, utilizing the Gazebo simulator for environment creation and testing. Deployed the system using Singularity for cross-platform compatibility and operated it on a TurtleBot equipped with an iDAR sensor for SLAM and navigation, powered by a Raspberry Pi 4 running ROS2.
- Fine-tuned a MobileNetV2 CNN to distinguish planets in the solar system. Expanded the dataset from 1,434 images to 50,000 through synthetic data generation using Gaussian blur, pixelation, and scaling techniques, improving model accuracy from 60% to 95% on the validation set.
- Employed Hough transform using OpenCV for star removal in captured images, utilising masking and dilation to extract and identify planets using the CNN model. The project achieved the highest mark of 90% among 45 groups, with real-life environment testing.