Michael Solotky

GitHub: MichaelSolotky
LinkedIn: michael-solotky

Gdańsk, Poland
✉ jawormichal128@gmail.com

Education

- **Master of Science** in **Applied Mathematics and Informatics**, GPA 3.90 / 4.0
Higher School of Economics : Faculty of Computer Science — Sep 2019 – Jun 2021
Joint programme with **Yandex School of Data Science**
- Bachelor of Science in Applied Mathematics and Computer Science, GPA 3.89 / 4.0
Lomonosov Moscow State University — Sep 2015 – Jun 2019
Faculty of Computational Mathematics and Cybernetics

Experience

- **Software Development Engineer at Amazon** — Aug 2021 – now
Alexa TextToSpeech C C++ Python
- **Research Science Intern at Yandex** PyTorch NumPy h5py Pyplot LaTeX — Sep 2020 – Jun 2021
  - **Performance validation** of existing methods for **uncertainty estimation** in more complex environments
  - Finding **theoretical foundations** for various methods of **uncertainty estimation** in **Deep Learning**
  - **Master's thesis**
- **Machine Learning Engineer Intern at Yandex** — Jun 2019 – Sep 2019
Machine Translation department TensorFlow MapReduce SciPy Pyplot
  - Conducted experiments to improve the quality and diversity of translations
  - Analyzed and found some basic mistakes that baseline approaches make
  - Implemented several successful models and inference techniques in Yandex's machine learning library with an ability to control diversity level
  - **Achieved statistically significant improvement in quality and diversity simultaneously** on Yandex's metrics, human evaluation compared to the baselines and on metrics commonly used in machine translation: about **1.1 growth of max-BLEU** and about **1.6 times self-BLEU diversity growth**
- **Software Engineer Intern at Yandex** — Jun 2018 – Oct 2018
Voice Technology department C++ Python MapReduce Protobuf
  - Implemented several methods of probability smoothing and their modification in language models for Automatic Speech Recognition
  - Conducted experiments on quality measurement to find the best model among all
  - Implemented an optimal algorithm for constructing n-gram language models in C++ using MapReduce, **which decreased wall time by at least 3 times and slightly increased quality measure** compared to baseline
  - Wrote a complete framework with a set of operations available from CLI

Projects

- **BigARTM** C++ Boost Protobuf Travis CI AppVeyor — Jan 2017 – May 2019
Open Source library for topic modelling with support of multiple regularization
  - Development and support of a tool for parallel calculation of pairwise word statistics such as frequency of mutual occurrence, PMI in large text corpora in conditions of low RAM
Wikipedia full-text processing takes less than 8 Gb of RAM now compared to at least 20 Gb needed before (code sample, documentation)

Technical Skills

- **Languages used at work**: C++, Python, C, Bash
- **Basic knowledge**: SQL, Perl, Assembly language
- **Technologies/Libraries**: MapReduce, Protobuf, C++ Boost, Make, NumPy/SciPy, Sklearn, Pandas, CVXPY
- **Deep Learning frameworks**: PyTorch, TensorFlow, Keras
- **Tools**: Git, Subversion, UNIX/Linux, Docker, Travis CI, AppVeyor, LaTeX