

EDUCATION

- **Master of Science** in Applied Mathematics and Informatics, GPA 3.9 / 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 Sep 2020 – Jun 2021
 - Performance validation of existing methods for **uncertainty estimation**
 - Finding **theoretical foundations** for various methods of **uncertainty estimation** in **Deep Learning**
 - **Master's thesis**
- **Software Engineering Intern** at [Yandex](#) Jun 2019 – Sep 2019
[Machine Translation department](#) TensorFlow MapReduce NumPy SciPy
 - 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 commonly used in scientific field metrics: **about 1.1 growth of max-BLEU** (maximum [BLEU](#) of generated translations) and **about 1.6 times n-gram diversity** (1 - [self-BLEU](#)) **growth**
- **Software Engineering Intern** at [Yandex](#) Jun 2018 – Oct 2018
[Voice Technology department](#) C++ 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 6 hours on octa-core intel core i5 8th gen, taking less than 8 Gb of RAM compared to at least 20 Gb needed before ([code sample](#))**

TECHNICAL SKILLS

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