CV: Sergey Ivanychev

👋 I'm a **Lead Data Engineer** and working at Constructor currently based in **Warsaw, Poland.** I enjoy building effective data infrastructure that makes colleagues productive, costs low, engineers confident and customers happy.

Contact me

- 🌏 ivanychev.org
- ✉️ sergeyivanychev@gmail.com
- 💼 linkedin.com/in/ivanychev
- ✍️ medium.com/@sergeyivanychev
- 👨‍💻 github.com/ivanychev
- 📕 bit.ly/sergey-ivanychev-cv

Main skills

- I code in Python, SQL, Go and TypeScript
- I build infra with AWS, Docker, Databricks and Kubernetes
- I build data pipelines with Spark, Luigi, Airflow and AWS Lambda

Education

- **Moscow Institute of Physics and Technology** — Bachelor's Degree in Applied Mathematics and Physics, GPA 8.8/10
- Yandex School of Data Analysis — Computer Science, Machine Learning and Data Engineering

Experience

Lead Data Engineer

Constructor, *Remote (December 2021 - Now)*

- Lead the Data Platform team (4 direct reports), design and develop the Constructor Data Platform

    - It receives, stores, processes various signals (backend logs, user behaviour events, database snapshots) — over 1 Petabyte of data
    - Provides data department tools for developing new pipelines and getting insights from existing data.
    - Exposes user-facing data using low-latency storage and services
    - Provides company management with cloud cost observability tools
- Develop data pipelines with **Apache Spark**, **Delta Lake**, **Databricks**, **AWS Lambda, Docker, Luigi** and **Python**. Cloud infrastructure with **CloudFormation** and **CDK**. Internal services with **FastAPI**, **React**, **AWS ECS**, **DynamoDB**. **Jenkins** pipelines for CI/CD. Customer-facing storage with **ClickHouse** and **DynamoDB**. Observability with **Prometheus**, **Victoria Metrics** and **Grafana**.
- Share technical knowledge across the organisation via regular internal newsletter (Tip of The Week). Share technical findings in the company blog on Medium.

Senior Data Engineer

***Joom**, Moscow, Russia (April 2020 — December 2021)*

- Deployed e2e data warehouse for our financial department in a separate AWS account in order to comply with banking security requirements. **(Scala, Spark, Flink, Data Studio, Airflow, EKS, RDS, Terraform).** Than involved deploying
    - An AWS EKS (Kubernetes) cluster with Apache Flink for streaming data ingestion and client monitoring, Prometheus/Grafana, Route 53, ALB.
    - EMR cluster with Apache Spark, Hive Metastore (using RDS) used as table metadata storage for Apache Spark, Zeppelin notebooks for data analysis, S3 buckets.
    - E2E ETL pipeline that is written in Scala/Apache Spark that is scheduled via Apache Airflow that involved data parsing, preparation of the detailed data layer, and the mart tables for BI tools such as Data Studio and Tableau.
- Python data platform **(Python, Scala, PySpark, Flask, Kubernetes)**. On top of the existing JVM language-friendly platform written in Scala I brought the support of Python tools for our analysts and ML-engineers. I built

    - Python API wrapper for Joom Core Platform API implemented in Scala/Spark + the standard data platform library.
    - A shared JupyterHub cluster that is used as a data exploration/analysis tool, AWS EFS.
    - Support PySpark pipeline development as first class citizen.
- Some other projects **(Kafka, Elasticsearch, BigQuery, S3)**
    - Data exploration UI: internal UI for exploring the data using the table definitions from Metastore: **React**, **ElasticSearch**.
    - Extended ETL processes, improved data platform, built multiple dashboards and data sources according to the business needs. Improved data ingestion from the backend systems via using Apache Kafka + Spark capabilities.
    - Performed cost optimisations that included the analysis of BigQuery SQL queries and its data access patterns, as well as S3 storage costs optimisations.

Software Engineer

Google, *Paris, France (November 2018 — April 2020)*

- Built and improved a high scale machine learning framework for YouTube **(C++, Spanner, BigTable, Flume, gRPC, Protocol buffers)**.
    - Made the single API for data extraction compatible with realtime (GRPC) and batch (MapReduce) mode via using advanced design patterns. This enabled us to benefit from microservice architecture.
    - Built a feature storage with Google Spanner that is available in both real-time and batch mode + feature metadata storage.
    - Built ETL pipelines using in-house batch processing frameworks and data storages.

Internships

- **Software Engineering Intern, Yandex** (May 2018 — October 2018)
    - Ranking quality team at Yandex.Zen. Improving ranking, data retrieval and experimental pipelines **(Java, MapReduce, SQL, Python, Spring)**. Enhanced monitoring of internal data processes **(Python, Flask, HTTP)**.

- **ML Engineering Intern** (October 2017 — December 2017)
  - Ads quality research team. Explored new ML approaches in ads-related metrics prediction and enhanced the inner ML framework used for model training and evaluation **(Python and its scientific libraries, MapReduce and graph computations, SQL)**
- **Software Engineering Intern, Google** (March 2017 — September 2017)
  - YouTube Trends Team. Increasing quality of Trending. Implementing additional infrastructure to gather required data and process it using Google technologies **(Python, MapReduce, Protocol Buffers, BigTable, Dremel SQL, Go, C++)**.

Open source

- **CatBoost —** machine learning method based on gradient boosting over decision trees.
  - Improved metrics serialization/deserialization in order to support reproducibility of the models used during training.
  - Added some more metric customization capabilities — users now can pass prediction border params that affects how regression turns into classification, alongside with custom descriptions of the metrics. This makes them more clear for the researcher during serialization.
  - Dramatically refactored the module that is responsible for metrics computation so that it's more extensible, easier to read and less error prone.
  - 3 PRs with nearly 6K LOC diff.

Education

- **University:** Moscow Institute of Physics and Technology
- **Degree:** Bachelor of Science
- **Principal Studies:** computer science, applied mathematics, machine learning.
- **GPA:** 4.9/5 or 8.8/10. Awards: Abramov scholarship (4 times, for learning progress and achievements, less than 10% of students receive it).
- **Additional Education:** School of data analysis, Intel iLab (advanced C and C++ programming)

Skills

- **Languages:** 🇺🇸 English (fluent), 🇷🇺 Russian (native), 🇵🇱 Polish (A1-A2), 🇫🇷 French (elementary)
- **Programming languages:** Python, SQL, Go, Scala, Java, JS/TS, C++
- **Technologies:**
  - **Computation**: Apache Spark, Apache Flink, BigQuery, AWS Lambda
  - **Databases**: ClickHouse, DynamoDB, Apache HBase / Apache Phoenix, PostgreSQL, Apache Hadoop, Apache HDFS, Apache ZooKeeper, BigQuery, Google Spanner
  - **Cloud**: AWS (EKS, ECS, EMR, S3, EC2, Lambda), Databricks, Docker, Kubernetes
  - **Data Science:** Python data science stack (numpy, scipy, pandas, scikit-learn), TensorFlow
  - **Frontend:** TypeScript, Bootstrap, React
  - **Backend:** FastAPI, Django, Celery, Nginx
  - **Miscellaneous:** Gradle, Apache Oozie, Yarn, Poetry