EURA NOVA

FACULTIES OF SCIENCE AND ENGINEERING

Internship Offers 2020-2021

For the latest information, please visit https://research.euranova.eu.
4th June 2020

CONTENTS

EURA NOVA

INTRODUCTION

EURA NOVA is a data-driven Belgian company founded in September 2008 and located in Brussels, Marseille, and Tunis. Our mission is simple: bring life to our customers' great ideas, by offering best-in-class services in data science, software engineering, and data architecture. To do so, we invest significantly in in-house expertise and state-of-the-art knowledge. In line with this course of action, we offer academic programs in collaboration with universities. These offers include bootcamps, master theses topics, research internships, and PhDs topics. See below for details.

OUR INTERNSHIPS OFFERS

This document presents internships topics supervised by our software engineering department or by our research & development department. Each project is an opportunity to be actively involved in the development of solutions to address tomorrow's challenges in ICTs and to implement them today.

The students will work in a dedicated **international** team of engineers **with diverse expertise in machine learning, graph theory, artificial intelligence, high performance computing, etc.**

They will keep EURA NOVA informed of the project advancement and share their ideas and challenges using the in-house knowledge management tool. We value continuous learning and teamwork. We love to have a good time together. For more information on our R&D activities, please visit our website at https://research.euranova.eu.

HOW TO APPLY

When you have gone through our internships offers, pick your favourite three. Draft a short text for each one, stating why you find it interesting and what you would do about it. Send us this statement, along with your CV at career@euranova.eu.

RESEARCH

APPRENTICE DATA SCIENTIST

Context

According to the Harvard Business Review magazine, data scientist is the sexiest job of the 21st century. Here, at EURA NOVA, we also think it is the coolest job. But what are we talking about? We are talking about ingesting diverse data sources, running distributed machine learning algorithms, visualising big graphs and so on. Sounds great, right?

Business Opportunity

At EURA NOVA, we leverage data to extract valuable insights for our clients. There are so many techniques to test and there is so much knowledge to digest. We hope that by sharing our knowledge with you, you will help us explore datasets much deeper.

Contribution

During the internship, you will be part of the data science team and will work in a stimulating environment. You will participate in data science projects using cutting-edge tools and techniques from the big data and machine learning domains.

APPRENTICE DATA SCIENTIST FOR DIGAZU

Context

digazu[1] is a batch and real-time big data platform developed by EURA NOVA. It is designed to fulfil customers' big data needs, such as real-time reporting, data integration, data science models industrialization… digazu is built onto cutting-edge technologies such as Kafka, Flink, Kubernetes, Kubeflow, HDFS, ensuring reliability, efficiency and scalability in production environments.

Objectives

This internship will focus on defining and building end-to-end use cases using digazu capabilities:

- Define a real-time data science use case
- Collect data
- Build a real-time data science model
- Collect data in real time
- Apply feature engineering on streamed data
- Apply the data science model in real time

This will allow you to get to know cutting-edge big data technologies, as well as to get familiar with data architecture concepts and challenges. It will also introduce you with the end-to-end process of leveraging big data use cases.

[1] https://digazu.eu/

APPRENTICE DATA ARCHITECT FOR DIGAZU

Context

digazu[2] is a batch and real-time data supply chain developed by EURA NOVA. It is designed to fulfil customers' big data needs, such as real-time reporting, data integration, data science models industrialization… digazu is built onto cutting-edge technologies such as Kafka, Flink, Kubernetes, HDFS, ensuring reliability, efficiency and scalability in production environments.

Objectives

This internship will focus on analyzing the market of data integration platforms and on comparing the capabilities of digazu with its main competitors, such as Informatica, Talend, Attunity…

This will allow you to get to know cutting-edge big data technologies, as well as to get familiar with data architecture concepts and challenges. It will also help you gain knowledge on the big data landscape and trends.

[2]https://digazu.eu/

OPTIMIZE DNN ARCHITECTURE FOR ADVERSARIAL CROSS-MODAL RETRIEVAL

Location: Tunis, Tunisia

Context:
Cross-modal retrieval aims to enable flexible retrieval experience across different modalities (e.g., texts vs. images). The core of crossmodal retrieval research is to learn a common subspace where the items of different modalities can be directly compared to each other. a novel Adversarial Cross-Modal Retrieval (ACMR) method, which seeks an effective common subspace based on adversarial learning. But, this method suffers from a high cost of training DNN architecture. How can we optimize the DNN architecture ?

Objectives:
The goal of this internship is to:

- Study the state of the art of Cross-Modal Retrieval.
- Optimize the DNN architecture .

MULTIMODAL REPRESENTATION LEARNING USING DEEP GENERATIVE CANONICAL CORRELATION ANALYSIS FOR AN ML APPLICATION

Location: Tunis, Tunisia

Context

Multi-view representation learning is concerned with the problem of learning representations (or features) of the multi-view data that facilitate extracting readily useful information when developing prediction models. Deep Generative Canonical Correlation Analysis is a method for learning nonlinear transformations of many views of data, such that the resulting transformations are maximally informative of each other. DGCCA is the first CCA-style multiview representation learning technique that combines the flexibility of deep representation learning with the statistical power of incorporating information from many independent sources, or views. So the main task is applying an ML task (classification for example) on the vectors projected by the DGCCA.

Objectives:

The goal of this internship is to:

- Study the state of the art of Canonical Correlation analysis.
- Develop Deep Generative Canonical Correlation Analysis for an ML task.

ENGINEERING

COMPARISON OF FLINK EXECUTION MODES

Context

Flink is a distributed computing framework, so one of its main feature is to execute a task of a job on a specific machine of a cluster. How the machine is selected is up to an orchestrator. In the cluster orchestrator world, Kubernetes has become the standard. Flink jobs can be executed in two modes either relying on Kubernetes to assign the task to a machine or relying on the internal Flink orchestrator process.

Technologies

Apache Flink, Kubernetes, Apache YARN

Objectives

Benchmark execution time and resource consumption for the following execution modes:

- Internal Flink orchestrator on YARN (Default)
- Internal Flink orchestrator on Kubernetes
- Kubernetes orchestrator

Using the comparison of the results, the goal is to challenge the assumption that one mode is better than the others, and find out which one.

A GITLAB CI EXECUTOR FOR DOCKER COMPOSE

Context
GitLab is an open source web-based DevOps lifecycle tool that provides a Git-repository manager providing wiki, issue-tracking and CI/CD pipeline features. Its CI allows to execute the build on different executors[3].

Docker is a containerization technology. Docker Compose is a Docker tool for designing and running multi-container applications. It is very convenient to run locally those applications.

Technologies:
You will work with the following technologies:

- Git and Gitlab for collaboration,
- the programming language Go,
- Gitlab runner[4], and
- Docker and Docker-Compose.

Objectives:
During the internship, you will develop a new Gitlab CI executor in Go that allows to easily run Docker Compose application defined in a given YAML file. That executor will:

- take a given YAML file as input to deploy the services and the build job,
- run Docker containers directly on the host,
- secure and isolate those containers from other jobs,
- support volume mounting,
- support as many Gitlab CI features (e.g. cache, artifact).

[3]https://docs.gitlab.com/runner/executors/
[4]https://gitlab.com/gitlab-org/gitlab-runner

IMPLEMENT A SCHEMA REGISTRY

Context
Apache Kafka is a messaging system that has been developed at Linked In and open sourced in 2014. A few of tools have been developed around it such as the Apache Kafka Connect.

The Confluent Schema Registry allows to manage the schema of Avro messages that are produced on Kafka topics. It is developed in Java and relies on Kafka itself to store its data, including the schema. It is a critical part when other system needs to know the schema of the messages going though the Kafka topics. It works well but it has a significant memory footprint because the JVM.

During the intership, you will design and develop a new schema registry that is compatible with the API of the Confluent one.

Technologies:
You will work with the following technologies:

- Git and Gitlab for collaboration,
- a modern programming language that would allow you to meet the objectives (e.g. Elixir, Rust, Go),
- a data store that would allow you to meet the objectives (e.g. Kafka),
- Kubernetes and Docker to operate it.

Objectives:

The goal is to develop a schema registry that:

- has a small memory footprint,
- is compatible with the API of the Confluent Schema Registry,
- is fault-tolerant,
- can scale to answer a high reading load, and
- can be extended to manage other kind of schema (e.g. JSON).

IMPLEMENT A KAFKA CONNECT

Context
Apache Kafka is a messaging system that has been developed at Linked In and open sourced in 2014. A few of tools have been developed around it such as the Apache Kafka Connect.

The Apache Kafka Connect allows to copy data between Kafka and other data storage. It is developed in Java and relies on Kafka itself to store its data, including the schema. It is a critical part when you want to interconnect Kafka with existing data storage system. It is built with a plugin system, one for each data storage system, so it can be extended to any one. It works well but it has a significant memory footprint because the JVM.

During the intership, you will design and develop a new Kafka Connect that is compatible with the API of the original one.

Technologies:
You will work with the following technologies:

- Git and Gitlab for collaboration,
- a modern programming language that would allow you to meet the objectives (e.g. Elixir, Rust, Go),
- a data store that would allow you to meet the objectives (e.g. Kafka),
- Kubernetes and Docker to operate it.

Objectives:
The goal is to develop a schema registry that:

- has a small memory footprint,
- is compatible with the API of the Apache Kafka Connect,
- is fault-tolerant,
- can scale to handle high throughput and/or low latency,
- can be extended for each data storage.

CONTRIBUTE TO APACHE KAFKA CONNECT

Context
digazu[5] is a batch and real-time data supply chain developed by EURA NOVA. Apache Kafka Connect is one of the technologies it is built onto. It is part of the open source project Apache Kafka[6] that is written in Java.

Technologies:
You will work with the following technologies:

- Git and Gitlab for collaboration,
- Java,
- Apache Kafka,
- Kubernetes and Docker to operate it.

Objectives:
The goal of this internship is to address digazu needs by contributing to the open source project Apache Kafka Connect. For instance:

- implementing new capabilities (e.g. start from given offset),
- fixing bug,
- contributing to existing open source connectors (e.g. JDBC, Debezium),
- improving the documentation.

[5]https://digazu.eu/
[6]https://kafka.apache.org/project

CONTRIBUTE TO ELIXIR KAFKA CLIENT: KAKFAEX

Context
digazu[7] is a batch and real-time data supply chain developed by EURA NOVA. Apache Kafka is one of the technologies it is built onto and its core is developed in Elixir. To interact with Kafka from the Elixir code base, we are usiung the open source client KafkaEx[8].

Technologies:
You will work with the following technologies:

- Git and Gitlab for collaboration,
- Elixir,
- Apache Kafka,
- Kubernetes and Docker to operate it.

Objectives:
The goal of this internship is to contribute to the open source project KafkaEx by:

- supporting Kafka API not yet supported (especially ones useful to digazu e.g. new administration of consumer group and offset),
- fixing bug (especially ones blocking for digazu),
- adding new client features (e.g. back pressure management, pooling), and
- improving the documentation.

[7] https://digazu.eu/
[8] https://github.com/kafkaex/kafka_ex

PATTERN RECOGNITION ENGINE: A FORMAL SPECIFICATION FOR COMPLEX EVENT PROCESSING

Location: Tunis, Tunisia

Context
Processing event streams is an increasingly important area for modern businesses aiming to detect and efficiently react to critical situations in near real-time. The need to govern the behaviour of systems where such streams exist has led to the development of numerous Complex Event Processing (CEP) engines, capable of detecting patterns and analyzing event streams.

Objectives:
Follow up on the latest progress on the LEAD project, your mission in this summer internship is to:

- Understand Pattern recognition engine and its performance requirement.
- Learn about the latest work that has been done in this project(scientific Paper and documented code)
- Run complexity analysis through different scenarios and evaluate the performance.
- Understand Flink state and time internal management, in order to enhance the engine capabilities in term of latency and throughput.

Requirements: Java, Flink, Data structures and complexity analysis

IMPLEMENT A TEMPLATE OF A BIG DATA FRAMEWORK CONNECTOR FOR A LEARNING AGENT

Location: Tunis, Tunisia

Context

Big data frameworks come with more and more configurable parameters. Although these systems come with a preconfigured default setting, it has been proved that the configuration should be varied to better meet the application's needs and provide a better overall performance.

That being said, performing the parameters tuning manually is not an easy task when it comes to finding the optimal settings for a hundred parameters, especially since a misconfiguration might lead to a performance deterioration. This is why we propose to perform the tuning automatically. To do so, the learning agent should be able to read and write the configuration to the target system.

Objectives:

During this internship, you will be asked to write a template of a connector to read existing configurations and write recommended ones to the system. You will apply and test the feasibility of your template connector on a Big Data framework You will work with the following technologies: Kafka, Flink, Java, Docker.

The goal is to:

- Develop a template of a connector to read the current configuration of the system, and write/apply the recommended configuration on the system
- Apply and test the template on a Big Data framework

GRAPH ANALYTICS WITH SPARK AND NEO4J

Location: Tunis, Tunisia

Context:
Modern information systems consist of a large number of sophisticated and interacting business entities that naturally form graphs. These graphs are used in a wide spectrum of application domains, ranging from social and information networks to biological and transportation networks. In an enterprise ecosystem, a multitude of graph technologies need to be integrated to efficiently offer graph analytics capabilities. In this internship, we focus on the integration of efficient graph processing using distributed data processing frameworks such as Spark and Flink, and graph databases such as Neo4j.

Objectives:
The goal of this internship is to develop an efficient graph OLAP analytics engine, that is seamlessly integrated with Spark and Neo4j. This engine will:

- Load and store data from/to a graph database
- Implement a set of graph analytics (mainly selection and aggregation) algorithms
- (Optional) Provide an interactive graph analytics dashboard

Requirements: Apache Spark, Scala or Java, Neo4j and JavaScript are a plus.

REFERENCES