MAMMALS aims to provide low-latency inferences by running—close to the end user—simple machine learning models that can also take advantage of a small (local) datastore of examples. The focus is on algorithms to learn online what to store locally to improve inference quality and achieve domain adaptation.

PROJECT DESCRIPTION

A machine learning (ML) model is often trained for inference's purposes. Inference does not involve complex iterative algorithms and is therefore generally presumed to be easy. Nevertheless, it presents fundamental challenges that are likely to become dominant as ML adoption increases and ML systems are ubiquitously deployed and need to make timely and safe decisions in unpredictable environments [16]. Big cloud providers, such as Amazon, Microsoft, and Google, offer their “machine learning as a service” (MLaaS) solutions, but running the models in the cloud may fail to meet delay constraints. As an example, recommendation systems, voice assistants, and ad-targeting need to serve predictions in less than 20 ms. Future 5G wireless services for connected and autonomous cars, industrial robotics, mobile gaming, augmented and virtual reality have even stricter latency requirements, often below 10ms and below 1ms for what is now called the tactile internet [15]. Such requirements can only be met by running ML prediction services at the edge of the network—directly on users’ devices or at nearby servers—without the computing and storage capabilities of the cloud. Privacy and data ownership also call for inference at the edge.

The current approach to run inference at the edge is to take state-of-the-art (SOTA) large ML models (often from the cloud) and generate smaller ones through compression or distillation [6]. MAMMALS will pursue a different direction. Its key idea is to take advantage of data availability at the edge (where data is usually generated) to compensate for additional computing constraints. In particular, we want to combine the decisions of a small ML model, e.g., a compressed neural network, with those of an instance-based algorithm relying on a local datastore, like k-nearest neighbors (k-NN). Instance-based algorithms can explicitly memorize rare patterns that are difficult to learn by simple ML models. Moreover, they do not require complex training and can efficiently incorporate new information. This activity builds on some recent findings showing that ML models can benefit from the presence of a local datastore or memory. Inspired by the (complex) memory-augmented neural networks [10, 11], some recent papers [9, 13, 12], have shown that the performance of SOTA neural networks can benefit from a memory storing a simple collection of examples, from which the most similar ones to the current input are retrieved to improve neural network inferences. These results are quite surprising as “in the machine learning research community it is generally believed that there is a tension between memorization and generalization” [4]. MAMMALS will exploit this synergy of model-based and instance-based learning to achieve more flexibility in adapting inference engines to limited edge resources.

Instances selection is a challenging task, and MAMMALS indeed focuses on designing online algorithms to decide what to store locally. Note that this corresponds to train the instance-based algorithm.

REFERENCES

Main activities

In the framework of the project described above, the postdoc can work on a combination of the three following aspects.

I. Design of online learning algorithms.

We plan to evaluate three different frameworks for learning online how to populate the local datastore.

1. Adapt existing caching policies like LRU, e.g., by inserting the content on the basis of its marginal utility (e.g., its contribution to inference quality). Ad-hoc policies in this spirit have been proposed to support image similarity search in [9] and in [8]. This framework leads usually to a combinatorial analysis with a focus on expected performance under a stochastic request process.

2. Study the problem as a discrete-space metrical task system (MTS) [2], where the state of the system is the set of instances in the datastore. Each state has a corresponding service cost (the loss of inference quality due to running a simpler model at the edge) and updating the datastore generates so-called movement costs. Competitive analysis is the common approach to study this setting.

3. When the set of possible instances is very large and roughly homogeneously distributed, at least partially on the topological properties of the space where instances lie. Whereas we are looking for collaborations with other research teams studying the topological and geometric structure of data, we will push a practical approach, starting from real traces. Many traces are available for recommender systems based on ML predictors. This application is particularly interesting for MAMMALS, as recommendations need to be customized to the user (a particular example of domain adaptation) and constantly updated to follow dynamic popularities of media contents or products.

II. Characterization of datasets' topological properties.

Which framework, among the three described above, is the most appropriate? The answer depends on how much we understand the large extent on the topological properties of the space where instances lie. Whereas we are looking for collaborations with other research teams studying the topological and geometric structure of data, we will push a practical approach, starting from real traces. Many traces are available for recommender systems based on ML predictors. This application is particularly interesting for MAMMALS, as recommendations need to be customized to the user (a particular example of domain adaptation) and constantly updated to follow dynamic popularities of media contents or products.

III. Prototype implementation.

We plan to provide practical evidence of the potential improvements from MAMMALS new algorithms in a simpler context. In many ML and information retrieval applications it is required to retrieve fast the k nearest neighbours (k-NN) of a given point in a dataset. When the number of dimensions exceeds 10, exact k-NN computation essentially requires to scan the whole dataset [17], so approximate indexing structures have been proposed and are currently implemented in libraries like Facebook Faiss [7]. Now, these systems can also benefit from a fast memory that stores a small subset of the whole repository. Managing this memory dynamically presents many of the challenges described above with the advantage of 1) avoiding the additional complexity of the interaction with the model, and 2) having a clear evaluation framework with well established benchmarks and performance metrics.

Skills

We are looking for one of the following profiles:

1) a candidate with solid analytical skills to design algorithms with strong performance guarantees,

2) a candidate expert on high-dimensional data analysis,

3) a candidate with hands-on experience on machine learning, able to reproduce state-of-the-art results like those in [12] and in [19].

Benefits package

- Subsidized meals
- Partial reimbursement of public transport costs
• Leave: 7 weeks of annual leave + 10 extra days off due to RTT (statutory reduction in working hours) + possibility of exceptional leave (sick children, moving home, etc.)
• Possibility of teleworking (after 6 months of employment) and flexible organization of working hours
• Professional equipment available (videoconferencing, loan of computer equipment, etc.)
• Social, cultural and sports events and activities
• Access to vocational training
• Social security coverage

Remuneration
Gross Salary: 2653 € per month