Software Engineering Assignment

Thibault Marette
marette@kth.se

1 Introduction

I am a theoretical computer science PhD, in the domain of algorithmic and my current research topic focuses on solution refinement. The main idea revolves around the fact that, in real-world applications, data analysis often involves an iterative optimization process, and solutions must be continuously refined in response to newly-collected or modified data. Typically, a solution is computed for the current dataset, but as new data becomes available, the solution must be updated to incorporate the latest information (see [1] in the context of data clustering). In addition to seeking a high-quality solution for a data-analysis optimization problem, it is also crucial to maintain stability by minimizing drastic changes from previous solutions. My research seeks to update smoothly a possibly outdated solution into a solution that fits the new data, without recomputing a new solution from scratch.

2 Lecture principles

One concept that echoed with me during the lecture was the principle of verification and validation. As stated above, I am working with solution refinement, which means that the objective function is given by the problem statement, and it is easy to decide if a new solution is better or worse than the original solution, since it suffices to compute the score of the current solution, and that we know the score of the initial solution. This makes the verification easy for this set of problem, but the validation is not covered this way. Validation would imply checking the sanity of the objective function (and the problem constraints), and is an interesting point of view to always keep in mind.

Another principle that I found interesting is Behavioral Software Engineering, which is the incorporation of social and human aspects to software

engineering. The core idea from this is to remember that software will be interacted with and made by humans. This is interesting to me as it is a component lacking in theoretical computer science: the major focus of the discipline is to find new ways to minimize objective functions. For instance, a recommander system will focus on finding new related by minimizing some distance function, without taking into account that humans will interact with the recommander system, and that measuring the impact of that system takes more than just a distance function.

3 Guest-Lecture Principles

The two concepts from the guest lecture that I will discuss here are the notion of problem space, input space, and their relationship. In my domain, we often talk about the problem space (or input space) and the solution space as defining the complexity of the problem and shaping which tools are appropriate to tackle the problem. For instance, a classic way to solve Integer Linear Problems is to relax the integer constraint into a continuous constraint. This changes the problem space as well as the solution space (by allowing variables to take more values than just 0 or 1) but interestingly enough, we can show than the solution space of the original problem and the relax problem overlap on some crucial points, that are the optimal values. This allows for a easier navigation in the solution space (since variable are now continuous), while guaranteeing than the global optimal solution are identical.

Moreover, a common technique in complexity theory is to alter the problem space of classical problem in order to come up with new and interesting problems. It is the case when, for instance, introducing fairness onto classical algorithms such as clustering: fair clustering is introduced as a variant of classical clustering, with the fairness constraint added ontop of it. This reduces the problem space significantly, but interestingly enough does not always result in problems easier to solve.

4 Data Scientists versus Software Engineers

It is difficult to fully agree or disagree with the ideas presented in these chapters. I partially agree with the authors that the fundamental difference between a data scientist and a software engineer is the set of skills brought by the difference in their background. I especially agreed the point of view to describe LLMs as software engineer component, and I could imagine in the

near future a portion of data engineers working on LLMs adopting software engineering practices for testing and validation for instance. However, I don't think it will be the only thing as research one LLMs need specialized theory people that will inevitably fall into the same pitfalls again. And while the "T-shaped" expertise is nice objective to reach, it is in practice pretty hard to attain.

The book argues that one needs a system-wide view over ML-enabled product, from data collection to model monitoring in order to safely and properly incorporate ML features in a product. This implies that both software engineers and data scientists need to cooperate in order to create reliable ML-enable products, but it does not imply necessarily that both roles will merge into one. Take for instance software developers using the agile method. Schematically, front-end and back-end developers have vastly different knowledge and method, yet they know how to communicate just enough to put their expertise to contribution for a working product on both ends. It is not far fetched to see similar organization emerging for ML-enable products.

5 Paper analysis

5.1 Yarally, Tim, et al. "Uncovering energy-efficient practices in deep learning training: Preliminary steps towards green ai." CAIN 2023.

This paper focuses on energy efficient practices in the context of deep learning training. The motivation is that more efficient training leads to lower energy consumption and hence a 'green' AI. This is achieved by focusing on two research questions. The first one is evaluating different training strategies with respect to their energy consumption, and the second one is to study how can neural network be compressed without losing too much accuracy. The main findings are that Bayesian optimization is the hyperparameter optimization technique that converges the fastest. Furthermore, the convolutional layers are, by far, the most energy hungry layers, and they should be removed as long as accuracy remains within an acceptable threshold. Unfortunately, no automated method was provided to achieve this, leaving it to be a manual case-by-case practice.

This paper is of interest for my research as we strive towards 'greener' algorithmics practices. The main idea is not to evaluate the performance

of algorithms according to a single metric, but also to monitor energy consumption and minimize it too.

This ideas is already present in LLMs, where shallower version of already existing models come into play, with being faster (and hence consuming less energy) and giving similar accuracy levels, how to train LLMs, that can have hundreds of millions of parameters. We can think for instance about 'chatGPT mini', which is smaller and hence more cost-efficient than the classical chatGPT model.

This paper can influence my research in an indirect way. While the techniques proposed here are irrelevant for my field, the concepts and the general point of view can be brought into the algorithmic field. A short term change is to add in the algorithm empirical evaluation its energy efficiency, while a long term search can be the study of energy-efficient algorithms.

5.2 Khadka, Krishna, et al. "A combinatorial approach to hyperparameter optimization." CAIN 2024

The core idea of the paper is to bring techniques from automated software testing to the field of hyperparameter optimization. They argue that the number of hyperparameters of ML models have been rapidly increasing, leading to overwhelmingly big search spaces, and traditional methods, such as random search or Baysian optimization are not sufficient anymore. That is why t-way testing, a software testing technique, is introduced in the context of hyperparameter optimization. The main principle of t-way testing is to find a set of relevant hyperparameters, and generate a t-way set with these hyperparameters. The model is then trained with all the different combinations of the hyperparameters, and the configuration maximizing a given metric is finally chosen as the best hyperparameter configuration.

This paper echoes with my research as, as aforementioned, I work with creating algorithms to solve theoretical problems. These algorithms usually have hyperparameters, to tune specific behaviors or algorithmic properties. Learning how to chose the best hyperparameters is empirically difficult, and it is important to have to know automated and efficient ways to tune them.

A crucial aspect of AI-intensive project is the hyperparameter tuning. If the hyperparameters are not correctly selected, it might result in a underperformance of the system. This tuning used to be done manually, but more and more, automated methods are emerging (such as the one cited in the paper). Any application using deep learning are subject to hyperparameter

tuning, that is why the paper's idea is very relevant to a broad range of applications.

In my research, I often have hyperparameters to chose. Having one more way to tune them is beneficial, and could be an automated way to select my algorithms hyperparameters during experiments in the future.

6 Research Ethics & Synthesis Reflection

To find papers, I focused on recent and highlighted (with the Distinguished paper Award Candidate) papers, where I could find a few keywords in the title that sparked my interest.

I found the 'green AI' paper to be quite misleading in its title, as they designate 'cost-efficient' AI as 'green' AI, where 'green' usually refers to some level of sustainability that is absent from the paper. Furthermore, creating more energy efficient neural network can provoke a rebound effect, where more energy efficient technology do not lead to a decrease of energy consumption, but in a more massive deployment of the technology (since it became cheaper), and this was not addressed at all in the paper.

To ensure originality of my work, I only studied the documents provided on the canvas page, as well as the papers online. no LLMs or exernal tools were used in the making of this assignment.

References

[1] Jon C Ergun et al. "Learning-augmented k-means clustering". In: *arXiv preprint arXiv:2110.14094* (2021).