P-4

Depending on the problem as well. I would say, some non-functional requirements might not be that important. For example, a portability of the system, you are going to develop a system that is going to run in an embedded system , of course you want something that is portable, of course you want a model that can run by a small micro controller. For example, if you are creating a model that is run on the cloud, portability is not an issue.

P-4

So coming back to your question, which non-functional requirement is less important, I will say its depends on application itself.

P-2

I don’t know, now I can’t see all non-functional requirements, it’s really tricky.

P-7

I would say maybe flexibility a bit or like reusability. It is not as important maybe. Because you have created the project then the, from my point of view few understand it is not going to be as easy just to slop out of blending new different data. You will still need to do some maintenance, you will need to prepare the data a lot, so that may not be as important anymore for machine learning models. Because at that moment there are other important non-functional requirements like efficiency and correctness.

P-9

No, honestly, I cannot think of any. I think rather it will be more non-functional requirements you mentioned this transparency, fairness, let us call them ethical requirements and data requirements, safety requirements would play a role but also security requirements which are if you would group of non-functional requirements, they will of course play a role as well. As an example, would of this possibility of the role you can trick a neural network by holding up some random pattern on a piece of paper and the system could recognize it as a stop sign for example it stops the vehicle. So now I don’t think there is any in the non-functional requirement that would play a lesser role.

P-8

In any software, yes, not very important in Machine Learning. In machine learning, you will ignore that, but in general software, it is important, but we have a lot of biases.

P-10

If we don’t have good data, if we don’t get it and if we cannot provide good data to Machine Learning, it will not give good results. Bad inputs will give bad results. So, we have to justify whether we should or should not adopt Machine Learning for some cases. In some cases, rule-based systems, decision trees will work much better, but nowadays there is hype that everybody wants to utilize the deep learning models, even it will work or not. Even as a researcher, it makes us crazy.

P-3

No, I don’t think so. I believe the full range of NFRs would apply to traditional software and machine learning enabled software as well.

P-3

Because ML analytical models, they are not used by end users, so directly they don’t contribute things like user experience. They are things those are used by overall software but don’t directly interact with end users. So, anything to do with end users usually not very center of importance when we talk about machine learning software.

P-3

Those which are less to do with customer.

P-1

Usually efficiency is not such a big deal but when it comes to ICU or close to patient, how the tool needs to respond it can be important.

P-1

Testability is not so important.

P-1

Flexibility right now not so important. If you need to scale up, you can do some changing, so we don’t consider that as much important thing yet.

P-1

The same with reusability. I think as AI is not so much mature yet so we are not considering it yet.

P-1

Same as interoperability.

P-1

Usability is not so much right now.

P-1

Efficiency is not so much yet at leas

P-1

Product revision and product transition are not so much important for AI as I think AI and Machine Learning tools are not that much mature as yet. I think this things comes later.

P-6

The usability is more related to front end part. Machine learning is more background component. If you need to be effective in machine learning, you want to collect the right information where the human is in the loop, it is not so important like traditional software.

P-6

Flexibility is less applicable for because it is quite scope-based application. The technology is possible, but the specific implementation that you have normally is quite oriented to the result.

P-6

If I have to export this non-functional requirement to my user, I will export only those parts which are meaningful for my user. For example, usability does not matter for the customer.