Speaker 1:

Okay. So first, I will ask you some questions about your background, so you can answer these questions briefly so that we can save more time for the other questions. So first, could you briefly describe your role in your team?

Speaker 2:

Yeah. I graduated from Computer Science Department and currently I'm working as software development engineer. Our team is working on search problems at Amazon. We are working on multiple projects, some of them are pure software development engineer projects, and others including deep learning and machine learning solutions to solve some problems. Yeah.1

Speaker 1:

So which kind of machinery models are you working on?

Speaker 2:

Yeah. We are working on both pre-trained models and we are building models from scratch.2

Speaker 1:

Okay. So can you tell me a recent time when you used a pre-trained model from an external model hub?

Speaker 2:

Yeah. I used one of them this year, basically from the beginning of the year to do some task in Arabic language. I used HuggingFace Hub.3

Speaker 1:

Okay.

Speaker 2:

Yeah.

Speaker 1:

Okay. So the next set of questions is related to how you slack a Pre-trained Neural Network. So here we are trying to understand the process that software engineers follow as they decide which Pre-trained Neural Network to reuse in their projects. So could you think about the last time when you choose a pre-trained neural network from a model hub? So how did you choose it? Can you summarize your decision making process for that?

Speaker 2:

Yeah. So I was looking for a model for Arabic language. Actually HuggingFace provides good filters to filter the models based on the language and the task. I start by looking at Arabic languages, the Arabic models sorry. I downloaded them and tested them on some data to make sure that I'm choosing the right model. Then I started by fine-tuning some of them, doing some tests on the test data. After that, I looked at the multilingual models on the hub, then also I fine-tune these multilingual models and compared with Arabic models. At the end, the process was somehow simple and easy, but there is no, let's say, predefined metric that says this model is better than this model. So I force to fine-tune all the models that I think they're promising and compare them on my own task.4

Speaker 1:

So is there a reason why you choose to use the models from the model hubs?

Speaker 2:

Yeah. Because I started with from scratch model. I tried multiple models like Fast Text. I built an LSDM from scratch and they didn't provide good results. So I just tried to use pre-trained models for Arabic and actually the results were better.5

Speaker 1:

Okay. So when selecting the model, do you care more about the model's performance than the architecture?

Speaker 2:

Yeah. As software development engineer with some knowledge of machine learning, I'm not interested that much in the architecture. I'm interested in the performance. If the model is doing well, then that's it.6

Speaker 1:

Okay. So do you think the Pre-trained Neural Network available in the model registries or model hubs accurately describes their behaviors?

Speaker 2:

Describes or behaviours in what manner?

Speaker 1:

Like the performance metrics in the documentation.

Speaker 2:

Okay. Actually the model author is responsible for doing that, but the model cards, for example, they're not comparable. So I can't log into model cards and say, oh yeah, this model A is better than that model B. Especially if the model authors didn't use the same test datasets, for example. As a solution of this problem, maybe the model hubs should create an automated work flow for example, that takes the model evaluates it on multiple data sets for that specific language or for multilingual languages to provide, let's say, a dashboard that compares all models, and I can choose the best based on my requirement or based on my problem.7

Speaker 1:

Okay. So to what extent do the discrepancies of performance metrics affect your decision making? So there can be discrepancies between the actual performances and the claimed performances in the model card.

Speaker 2:

Yeah. Based on my experience, when I start looking at the Arabic models, Arabic pre-trained language models, the model that shows better results on some datasets wasn't doing well on my task, while the model that is not performing well in the same task or the same list that showed better results on my task. But I don't know if this answers your question, but again, I need to test the model to get the answer.8

Speaker 1:

Okay. So do you usually to re-train the model or validate the model and make sure that the model matches the claims?

Speaker 2:

Actually no, because this is not a simple task, so I need to find the test set and download it, and pre-process it as described in the paper or they model card and pass it to the model, write the evaluation script. And all of this is not a simple task, so I will trust the model card.9

Speaker 1:

Okay. So to what extent does robustness of the models affect your decision?

Speaker 2:

It's the first factor. If my task is an offline task, for example, doing pre computation of some values and bought them as an offline data deficit in production. But if I want to deploy the model in production and use it in online productions, then I will look more into the model size and its performance in terms of latency and so on.10

Speaker 1:

Okay. So to what extent does the explainability of the model factor decision?

Speaker 2:

Extendability?

Speaker 1:

Explainability.

Speaker 2:

This is a very important thing for the leadership because they don't like the black box thing, it is a black box thing, official learning. But most of the models are not explainable. I know that we can do like the attention maps for language models and do the same thing of similar thing, the images and so on, but it's not scalable. You need to do it example by example and so on. So for me, it's not that important again, because I think it is related to the science part or then the engineer part. But if someone asked for it, I will go with available options, attention maps, and masks in images, science on.11

Speaker 1:

Okay. How frequently do you fine-tunes the models?

Speaker 2:

Yeah. In my daily job, I'm not that much. But outside of the job of that, as the job in the open source community, we are working on multiple projects. So maybe on monthly basis, I'm fine-tuning the model.12

Speaker 1:

Okay. So do you think the lack of fine-tune ability is a problem when reusing a Pre-trained Neural Network?

Speaker 2:

The lack of fine-tunability in terms of...?

Speaker 1:

Like, whether a model is fine-tuneable?

Speaker 2:

Actually, I didn't face that because all models provided in HuggingFace at least are fine-tuneable and this is an easy task to do. But for example, when I worked with TensorFlow Hub, I didn't fine-tune anything from their hub because I think it's possible, but I didn't find documentation, or I didn't search enough on how to do that from TensorFlow hub.13

Speaker 1:

Okay. So is there any other challenges you faced before when selecting a pre-trained model from the model hubs?

Speaker 2:

Actually, the experience is very simple in HuggingFace. Again, the issue is how to compare the models. Other than that, there's no issues. Maybe the issue or the thing and they solved it somehow that there are models like big models. So you can download them in the fast way and start fine-tuning or play with them. So they provide spaces now, and you can write code on them. So you don't need to use your machine. So I think this issue also they solve it.14

Speaker 1:

Okay. So then the next set of questions are about the depending software attributes. So we'd like to learn about what sort of information is useful to engineers who use a pre-trained models. So I'd like to share some definitions here. So this are the metrics from the npm, which is a platform for the JavaScript packages. And you can take a look at the definitions or the first several sentence for each attributes here and let me know why you are ready.

Speaker 2:

Okay.

Speaker 1:

So what do you think would best help your team's lack of Pre-trained model from the model hubs?

Speaker 2:

Actually in the current model hubs set up, I think the quality measures are important because some model cards are providing such information, but first not all of them second is not comparable. It's based on the author. And you need to navigate to each model card and find if this information exists or not. So you need to compare it manually. There is no like predefined thing that says, this is the place that author should put their task, let's say metrics. And this is the place that, where they put some code to test, model and so on. So there's no like predefined sections.15

Speaker 1:

Okay. So then I will show you some new attributes, which record different specific attributes. And I will ask several questions about each of them separately. So first for the provenance here, we mean a measure of model lineage or traceability. So some examples here are like, whether the model card contains link to the paper, like [inaudible 00:14:13] or like whether they provide the data set or architecture of the model. So can you think about a time when you met the provenance problem when using the pre-trained models before?

Speaker 2:

Yeah. The length of the paper is very important and I think it should be like redefined input in the model card. The model hub creators should ask for it. The [inaudible 00:14:44] page also should be available open-sourcing the code because we can like understand architecture in a more understandable way by reading the code. Also external side is not mandatory because not all models or not, but all author [inaudible 00:15:09] to build and try and do all of that for their models.16

Speaker 2:

But I think the most important thing is the dataset defining clearly what's set they used, how they pre-processed it, maybe providing a link to the version that they trained the model and tested the model on it, how they segmented data in train test and validation and all of that. Because without of this knowledge, we will not be able to compare this model with any other work. And this is a big issue because some of the authors of the datasets, they didn't split it in train test and validation, for example, and others didn't process it. So you need to process it for your task and all of these steps are not comparable. If we are not working on the same exact dataset, then our results are not comparable.17

Speaker 1:

Okay. So what do you think would be useful to know beforehand in order to solve the provenance problems?

Speaker 2:

Sorry, can you come again?

Speaker 1:

What do you think would be useful to know beforehand to solve the provenance problem? I mean, are there any other things we can put here?

Speaker 2:

Okay. Other of these depth sets or attributes, I can't think of anything else now. Maybe the model hubs can provide some graphing feature that allows the model authors to graph their architecture. And this graph can be dynamic and we can play with it. And like, for example, opening nodes and closing nodes and understanding the architecture, this would be a very good thing instead of opening the paper and finding the image that explains architecture and so on.18

Speaker 1:

Okay. So then we were talk about the reproducibility. So here, we mean the ability of departing practitioner to produce the same accuracy and same training or evaluation result from pre-trained model as defined in the paper source code or the engineering groups. So can you think about a time when you met any reproducibility problems? Could you tell me what were the challenges you met?

Speaker 2:

Yeah, in our graduation project, we were working on Arabic Text Diacritization and there is no single paper for Arabic Text Diacritization that was providing the model, the data, and you can test it easily and reproduce the results easily. So for each paper, we was like downloading the data, running the model on the test data set, get the results. Some of the models are using frameworks that are not used now or did frameworks. So you need to set up these things and this took days and actually weeks to do the setup and test these things. So during the Reproducibility thing is not easy.19

Speaker 2:

So the model hubs now are simplifying the process, especially on HuggingFace because you just load the model and you can start send inputs and you'll get outputs. But there is a point where the authors are doing some processing with the output. So is this processing like similar to what I'm doing to reproduce the results? So providing a script from the authors and put it as an attribute in the model card, like this is the script that we use to like, get the metrics of our model would be very helpful. Also, reporting some like hardware related metrics, like the memory footprint, the model latency, and so on. On a specific hardware, it'll help me to decide whether I will use this model or not.20

Speaker 1:

Okay. So the last attribute here is a portability. So here we defined as a, whether engineers can easily take a pre-trained model and reuse it in a different environment or a different software project. So can you think about a time where you met any portability problems? And could you tell me the challenges you met?

Speaker 2:

Yeah. While fine-tuning the Arabic model, for the task I was talking about and [inaudible 00:20:38], we faced some issues related to the model size. So we decided not to deploy it in production. We decided to go with offline solution. Doing that, like prevents us from getting the full benefit from the model. And this is because the model is big. It will increase the latency of the request. So again, the latency, the energy consumption should the model will be able to work on CBU, or should we use CBU for it? All of these attributes should be like provided. And maybe the model hubs could automate something like creating other formats of the model automatically like [inaudible 00:21:40] and optimizing the model. I think these things could be automated and provide a version that could be run on CBU instead of GBU and all of that. As I test the model, I can now deploy it easily, other than like, I test the model and I'm happy with the results. Now I need to quantize the model and do all of that job manually.21

Speaker 1:

Okay. So the last question for this part is except for this three attributes, do you think there are any other attributes can be helpful for the pre-trained models?

Speaker 2:

I think one thing that can be helpful especially in the industry. I know they are putting the license of the model, but I think they should link the license of the data also, because if the model let's say is MIT and the data is GMU version three. So I don't know what's the legal state here, but we should know both licenses to be able to decide whether we can use this model or not.22

Speaker 1:

Okay. So the last set of questions about the trustworthiness of the models. So we are trying to understand how the appreciation models shortcomings affect the engineer's ability to rely on and they reuse them. So, which aspects of the pre-trained model do you assume trustworthy?

Speaker 2:

The first thing is the affiliation for sure. So if Google, Facebook, Amazon are publishing a model, then I will trust that model. Actually, not all the time, these models are good and providing better results, but you trust the name. This is the first thing. The second thing is the well documentation of the model card. If the model card tells me as much information that as I need, then I would trust that model, especially about the testing process and dataset and so on. The third thing is the open-source mentality. If the model authors are open-sourcing their model, their code, dataset that they are using all of that, then I will trust that model because I can see all of the things. I can train the model from scratch. I can see the code. I can do whatever I want with it. So this three attributes can guide my decision or guide my trusting score for that model.23

Speaker 1:

Okay. So have you found any discrepancies between the claimed pre-trained model and the downloaded version? So for example, the accuracy, latency or architecture of the models.

Speaker 2:

Actually, I didn't face something similar to that.24

Speaker 1:

Okay. So, if there are some models has discrepancy problems, so to what extent do you think the discrepancies are acceptable?

Speaker 2:

I think if the misinformation is about, let's say the model size or something related to the hardware, then this is acceptable. Because if the author say that the model is 1.5 gigabyte size, and it's two gigabytes then, or they say that it's latency is 1.5 millisecond, or so sorry, 1.5 seconds. And I test it locally on the same hardware, and it is two seconds. Then both of them will not work in production for example, at least in the cases that I faced. But if the mass information is in the accuracy, the dataset, then this is a big issue because I can't do all the tests that the authors did on the test steps. And if there is misinformation in the test they used and the pre-processing steps that they did, then I will not be able ever to reproduce the results. So I think this is more important. Yeah.25

Speaker 1:

Okay. So do you think the discrepancies of the models will have significant impacts?

Speaker 2:

I think, yes. Because any misleading information or any, let's say mistake in this information will lead to something wrong. I can remember, I think it's related to first text or something like that when they reported in their website or in the paper different. No, it's about GPT-2. I think they reported in the paper, the number of fiber parameters or the number of parameters in the model. They reported a wrong number and they corrected that on their report. So this, while reading the paper, maybe I will get the attention to use that model because it's higher parameters or parameters is small. And I think it's lightweight on the hardware, but when I download it and start working with that, I surprised with the huge size of some of it. So this misleading information are very critically.26

Speaker 1:

Okay. Thank you very much. That's all of my questions. I stop recording now.

**Annotations**

1 Subject is a Dev Engineer at Amazon

2 Subject works on both pretrained models and models built from scratch.

3 Subject used HuggingFace to get a model for a "task in Arabic"

4 In order to select a model subject filtered models on huggingface by language (arabic and multilingual)

Subjected downloaded a few models, finetuned them, tested them then compared them.

5 Subject used a model hub model because they worked better than his from scratch model

6 Subject is not concerned with arcitecture, only performance

7 Model documentation is not compareable since authors use different tests.

A solution would be automated tests that are run on all models allowing them to be compared.

8 Subject had to test the model to make a decision

Could not reply on documantation

9 Subject never validates the model card

Because validation is "not a simple task"

10 Robustness is "the first factor" if the model is offline.

If deployed to producation latency and moel size is amound variables that are more important

11 Leadership does not like "the black box thing"

Explainability is not important for the subject

If someone requried it, subjct would use attention maps, masks in images and other avialable tools.

12 Subject does not finetune models for work but does for open source projects

Subject finetunes on a monthly basis

13 Finetuning is easy in Huggingface

Finetuning is hard in TensorFlow hub due to lack of documentation

14 Subject has not experince finetuning issues with huggingface

15 Quality measures would be helpful for pretrained models

Sometimes metrics are missing

Metrics can not be compared accross models

16 Length of paper is imporant and should be included in the model card

Rest is inaudible

17 Most important thing for providence is definind data set, how it was pre-processed,tested, segmented, validated, etc.

Differnet authors process data differently so it can't be compared.

18 A graph of the model archetecture would be useful to have

Graph should be dynamic and users should be able to "play with it"

19 Subject attempted reproduce test results as a graduation project and it took weeks to setup and test.

Reproducibilty is not easy

20 Hardware related metrics would be useful.

Test/processing scrips would be useful for reproducibility

21 Subject gives an example of model size effecting protability

Model was used offline, increasing latency

Model hubs could automate creation of other formats of the model for different hardware - this would make deployment easier

22 In addtion to the three atributes, licensing should be made explicit for industry users.

23 Afilitation is the first thing concidered for trustworthiness

Documentation is #2

Open source mentality is #3 - described like transparency

24 Subject did not oberseve discrepancies between claimed results and the downloaded model results.

25 Subject gives 3 example where discrpancies are unacceptable:

model size: 1.5 vs 2GB

latency: 1.5 vs 2 secs

accuracy: no example given but issue because results can not be reproduced

26 Discrepencies will have significant impact

Example of a model falsly advertized to be small

Subject was suprised by models large size