**Interviewer 2:** [00:00:00] Okay, here we go.

**Interviewer 1:** [00:00:02] So just could you confirm it on the record that you allowed us to record this interview and transcribe it?

**Interviewee:** [00:00:08] Sorry. I didn't get the last question.

**Interviewer 1:** [00:00:14] Could you confirm it on record that you allowed us to tape this interview and share the transcribed version?

**Interviewee:** [00:00:20] Yes, yes.

**Interviewer 1:** [00:00:22] Okay. So I will try, I will start by explaining the general objectives of this interview. So what we're doing is we are conducting a study and we are trying to create a taxonomy of the problems that developers encounter when they develop systems that use deep/machine learning. So as part of this study, we have analyzed GitHub commits/issues and StackOverflow discussions that are related to deep learning frameworks such as Tensorflow, Keras and Pytorch, but we think that these sources of information cannot be representative of all the issues that developers face and that is the reason we're conducting these interviews with developers. So what we are interested in is like the bugs/problems/challenges you faced when developing deep learning or machine learning systems. So, but first, I would start with general questions, some info about your background. Could you tell us what is your current position at your current job?

**Interviewee:** [00:01:23] I'm currently an assistant professor at [university: removed for anonymity].

**Interviewer 1:** [00:01:30] Okay, thank you. So could you tell us about your overall work experience and specifically your experience in deep learning or machine learning?

**Interviewee:** [00:01:40] Okay. I have a PhD in statistical signal processing. I have background in statistics and machine learning. I am applying machine learning and, in general, artificial intelligence techniques to network traffic analysis.

**Interviewer 1:** [00:01:57] Okay. So what type of deep learning networks have you developed or implemented? And by that I mean supervised/unsupervised/reinforcement learning and etc.

**Interviewee:** [00:02:11] Supervised. And we are also dealing with unsupervised.

**Interviewer 1:** [00:02:15] Okay, and what problems were trying to tackle using deep or machine learning networks? Like image classification, speech recognition?

**Interviewee:** [00:02:24] No, we are doing network traffic analysis.

**Interviewer 1:** [00:02:28] Okay.

**Interviewee:** [00:02:29] In detail, traffic classification and prediction.

**Interviewer 1:** [00:02:33] Okay, I see. And which programming languages and frameworks have you been using so far?

**Interviewee:** [00:02:40] We mainly use Keras with Tensorflow as the backend.

And

**Interviewer 1:** [00:02:46] as the programming language you use Python, right?

**Interviewee:** [00:02:49] Yes. Sure.

**Interviewer 1:** [00:02:50] Okay. Thank you. So as I mentioned before we have one general question for this interview, which is the problems/bugs/challenges that you faced when developing these systems. So maybe you could start from there and tell us whatever comes to your mind about this.

**Interviewee:** [00:03:10] Probably, the problem, I mean the problem is that when you have some some issues, some coding issues, you can find a lot of resources through the internet. But many of times these resources are not very [inaudible] and also the problem is that many of this guys that attempt to use technique, usually do it without a decent background. So, the type of answers are merely programming. I mean the type of answer that you would expect from a programmer, okay?

**Interviewer 1:** [00:03:55] Are you here?

**Interviewee:** [00:04:04] Yeah, I am still here.

**Interviewer 1:** [00:04:07] Do you have anything else to add?

**Interviewee:** [00:04:11] Sorry?

**Interviewer 1:** [00:04:12] Do you have anything else to add?

**Interviewee:** [00:04:14] No.

**Interviewer 1:** [00:04:16] Okay, so maybe I could ask you some questions. So have you used existing data sets to train your network or did you collect your data on your own?

**Interviewee:** [00:04:29] We both used existing datasets, but also collected our dataset in some specific application.

**Interviewer 1:** [00:04:39] Okay, so could you tell us how did you collect your data?

**Interviewee:** [00:04:44] Regarding the collection, we collected this in our laboratory.

**Interviewer 1:** [00:04:53] Sorry, you collected it where?

**Interviewee:** [00:04:56] In our laboratory.

**Interviewer 1:** [00:04:59] Okay, so could you give us some details?

**Interviewee:** [00:05:02] Yeah, we run the collection campaign in which students came and we got the traffic generated by mobile devices that were provided by us to the students.

**Interviewer 1:** [00:05:15] Okay, so you generate traffic using some app and then you train the network so that it can detect which app has generated it?

**Interviewee:** [00:05:27] Yeah, exactly.

**Interviewer 1:** [00:05:28] Okay. So did you pre-process the collected data in any way?

**Interviewee:** [00:05:33] Yes, both at the network level, but also at the, I mean, at the program level which means doing some standardization of datasets and so on.

**Interviewer 1:** [00:05:52] Okay. So do you remember any problems related to training that you have faced in the process?

**Interviewee:** [00:06:06] I don't have any. I don't remember any particular issues.

**Interviewer 1:** [00:06:15] Like did you have cases when you did not have enough data or the data was wrong in some way, like wrong labels.

**Interviewee:** [00:06:23] Yes, that's the main problem. Usually you don't have enough data, because in our problem the key fact is to have human-generated data, which is not an easy thing to do.

**Interviewer 1:** [00:06:41] So, do you try to tackle this problem in any way, like using data augmentation or do you just try to get more data?

**Interviewee:** [00:06:52] We worked on both. I mean we are planning and we are collecting more and more data, but we are also doing, I mean we are studying and we are implementing some sophisticated methods to cope with imbalanced and scarce data.

**Interviewer 1:** [00:07:14] Okay, could you tell us about the problems that imbalanced training dataset creates for you and the scarce dataset creates for you?

**Interviewee:** [00:07:23] Yes, this is one of the main issues in training deep learning networks. And usually we cope with this problem with cost-sensitive learning or undersampling and oversampling, depends on the occasion.

**Interviewer 1:** [00:07:44] I see. So you said that you have also used existing datasets. Could you tell us which ones did you use?

**Interviewee:** [00:07:51] There are some datasets by the Canadian Institute for Cybersecurity and also there is a dataset the was released by Dallas University in Canada.

**Interviewer 1:** [00:08:07] So when working away with existing datasets, did you have any problems when using the existing ones?

**Interviewee:** [00:08:16] Yes, the main problem is that most of the times the dataset is not formatted or more specifically does not provide the inputs that you want to use. Because, the dataset are often pre-processed only to provide some specific inputs.

**Interviewer 1:** [00:08:42] Okay, would you give us like more details? What kind of problems related to the format of their input do you have? Is it wrong type, wrong dimension?

**Interviewee:** [00:08:55] No, not wrong type of wrong dimension, but to the limitation of the way the dataset is exposed publicly. Which means that, for example, you have that for a given instance, you don't have all the inputs that you may draw out if you were collecting the dataset by yourself.

**Interviewer 1:** [00:09:19] Okay, and it did how did you deal with this problems?

**Interviewee:** [00:09:23] We collect our dataset.

**Interviewer 1:** [00:09:26] Ahh, ok, makes sense. So in the end you were not able to use this existing datasets for training, right?

**Interviewee:** [00:09:36] No, we were able to use them, but we were limited by the type of method that we were, that we could apply and we could test.

I see.

**Interviewer 1:** [00:09:48] Okay. So, maybe we could talk about model structure. So have you ever faced problems related to the wrong model structure? And by that I mean about the number of layers, types of the layers and etc. And if yes, what were the problems?

**Interviewer 2:** [00:10:06] Or maybe the model type itself.

**Interviewee:** [00:10:09] Yeah.

Yeah, probably not to the wrong architecture as whole, but more usually to the wrong type of layer, because usually people have, in our field people have applied like type of layers which were not suited for the type of input which they are processing. Like using convolutional layers for non-spartial data or non time-related data.

**Interviewer 1:** [00:10:51] Sorry, non what related data?

**Interviewee:** [00:10:55] Time-related data.

**Interviewer 1:** [00:10:58] Okay, time-related data.

**Interviewee:** [00:11:00] I mean data which have time correlation.

Okay. I see.

**Interviewer 1:** [00:11:04] Have you ever faced problems related to hyperparameter tuning?

**Interviewee:** [00:11:09] Yes, and usually we are trying to perform an hyperoptimization of the parameters to deal with this problem.

**Interviewer 1:** [00:11:23] So do you use some automated tools for that?

**Interviewee:** [00:11:27] Not yet, but we are currently working on this.

**Interviewer 1:** [00:11:35] Okay, so you define your hyperparameters manually then, right? So could you remember any hyperparameter which value you have changed and it improved the performance or the accuracy of your model significantly?

**Interviewee:** [00:11:51] Like the number of layers, the number of neurons inside the layer, for convolutional layers the extent of the filter and the stride(?) and, for example, the amount, the type of pooling that you must form when you use convolutional layers, or like the size of the memory in recurrent neural networks and so on.

**Interviewer 1:** [00:12:17] Okay, did you have problems with hyperparameters like I don't know batch size, number of epochs, activation function?

**Interviewer 2:** [00:12:26] Learning rate.

**Interviewee:** [00:12:26] It was also those parameters, the activation function, also the type of learning algorithm and there are many other parameters that you have to tune to improve the performance. But the main parameters that will boost your performance are usually those related to probably to the architecture, so the number of neurons and the number of layers. I mean, when you change the training algorithm there is only a marginal improvement.

**Interviewer 2:** [00:13:06] And did you ever had any problems/bugs related to dimensionality of the layers?

**Interviewee:** [00:13:12] Yeah, because that is what I said when I spoke about the number of neurons.

**Interviewer 2:** [00:13:20] Could you please give some details about this?

**Interviewee:** [00:13:26] What do you mean by details?

**Interviewer 2:** [00:13:28] Could you, please, tell us about some specific case you had which you can remember.

**Interviewee:** [00:13:34] I mean, for example, in our case, network traffic analysis, the type of inputs may be very large, so you could try some type of layers which have a high number of neurons or a high number of filters, but usually this is not always something that you can do, because the hardware has limitations, you have to cope with this trade-off of having a large network and also having a network which trains for a long time.

**Interviewer 2:** [00:14:14] Thank you.

**Interviewer 1:** [00:14:17] Okay. Thank you. About loss functions. Do you usually use a predefined or a custom-written loss function? And did you have any problems because of the wrong loss function or wrongly implemented loss function?

**Interviewee:** [00:14:31] We usually adopt built-in loss functions, for both regression and classification problems. We are dealing with the use of more sophisticated loss functions or for dealing, for example, with class imbalance. But I have never had problems with wrongly implemented loss functions. So we usually use the predefined functions by Keras, so they are ok.

**Interviewer 1:** [00:15:03] But but you're looking for the more custom written ones because the predefined ones are not doing their job very well or it's not appropriate for the type of problem you are addressing?

**Interviewee:** [00:15:18] Yes, because the problem is very hard, so the type of loss function, the type of standard loss function does not allow you to solve the problem properly, so you're not getting the desired performance.

**Interviewer 1:** [00:15:34] Okay, and about the hardware, where do you train your models and do you remember any problems that you had because of hardware?

**Interviewee:** [00:15:48] I'm not dealing with this personally, but the main limitation is that we are currently training our network on non-dedicated hardware, the main problem is the time, you have to wait to see if your network is performing well. So what I can say is that the main issue is that these kinds of networks are very, I mean they require a long time to train. So the main problem is related to having a dedicated hardware, which allows you to speed up the retraining process.

**Interviewer 1:** [00:16:40] Did you try to take any other steps to anticipate this problem like training in batches and etc. Or do you just think that you should better hardware?

**Interviewee:** [00:16:51] Sorry, I did not get the first part of the question.

**Interviewer 1:** [00:16:54] Yes. I was asking if you are taking any steps to anticipate this problem of long training time or large training data, like training in batches or anything?

**Interviewee:** [00:17:09] Yes, we are. We are doing this by limiting the number of epochs for the training and also by doing some, by imposing some exit condition, like the early stopping method.

**Interviewer 1:** [00:17:33] Okay. I see.

**Interviewer 2:** [00:17:37] A little question. So when you train your models do you use GPUs?

**Interviewee:** [00:17:45] We are not currently using GPUs, but we are planning to buy them, to train our networks.

**Interviewer 1:** [00:17:57] So your ground truth for your training, like when you do evaluation, is what? Is it the collected data where you have generated traffic and you know which application is it, right?

**Interviewee:** [00:18:12] Yeah, because we asked each user to play with the specific app, so we know in advance which will be the ground truth.

**Interviewer 1:** [00:18:22] So when you generate traffic, can you use that data as it is, or do you have to apply some steps to make that data useful for training? Like do you have to extract some features? Do you have to clean something?

**Interviewee:** [00:18:37] No. No we have to extract the meaningful input data to feed the deep learning architecture, because the traffic is made of the packets which are encapsulated, they're different layers of the stack, so we have to draw out the information which are useful for our task and also avoiding using some information which may bias the performance evaluation, because the collection in our case depends on where you are gathering your traffic, so like using some parts of the package, so to inflate the performance, like using IP address.

**Interviewer 1:** [00:19:29] So, how many different applications have you used for your training data?

**Interviewee:** [00:19:37] What do you mean by application?

**Interviewer 1:** [00:19:40] Like you have different applications that generate traffic and you're trying to identify which application this traffic belongs to?

**Interviewee:** [00:19:48] Not many, not many.

**Interviewer 1:** [00:19:51] So yeah, how many different applications?

**Interviewee:** [00:19:53] Yeah, we are dealing with 40-50 applications, different applications.

**Interviewer 1:** [00:20:01] So have you tried to have more than that?

**Interviewee:** [00:20:04] Yes, we are. We are limited by the collection phase, because we have to get more people to gather and enlarge our dataset.

**Interviewer 1:** [00:20:19] So do you remember any case when you did your training, you get some level of accuracy, then you take some steps and your accuracy improves. And if yes, could you tell us what are those steps?

**Interviewee:** [00:20:36] No, sorry. I didn't get this. Can you repeat?

**Interviewer 1:** [00:20:39] Yes, of course. So I'm telling that let's say you have your training data. You have your model structure. You have put some hyperparameters and you train your network and you get some accuracy, right?

**Interviewee:** [00:20:54] Okay.

**Interviewer 1:** [00:20:55] So do you remember that, like after doing this, you took some steps. I don't know you changed something and then your accuracy has increased. If yes, could you tell us what those steps are?

**Interviewee:** [00:21:09] Like taking into account the imbalancing loss function, like increasing the number of layers, like changing the type of layers or changing the type of input, the dimensionality of input, changing the activation function.

**Interviewer 1:** [00:21:30] About the activation function. Do you remember which activation function you have changed to which one, for example?

**Interviewee:** [00:21:42] Yeah. Usually we switch from classic sigmoid problem or hyperbolic tangent to rectified linear unit or more sophisticated versions of them.

**Interviewer 1:** [00:21:55] Okay. You also said that you have also worked with unsupervised systems. So I am guessing that this network traffic one is supervised. So could you tell us more about these projects that you've worked on which was unsupervised learning?

**Interviewee:** [00:22:17] Yes. We are dealing with the network anomaly detection, in which you have during the training phase all the normal data and you don't have access to attack data.

**Interviewer 1:** [00:22:31] Okay, and then how do you evaluate that one? You generate a test set that has an attack data?

**Interviewee:** [00:22:40] Yes, the test set both contains the benign and the attack data.

**Interviewer 1:** [00:22:46] So how do you collect that attack data?

**Interviewee:** [00:22:51] In this case, we are using a dataset that is publicly available. So for this problem, we are not doing collection. Because it is very very hard to emulate a network attack.

**Interviewer 1:** [00:23:09] Okay and did you ever had problems using those pre-existing datasets or those ones are ready to use, unlike the ones you told us about before?

**Interviewee:** [00:23:22] Yes, mainly the same, the way the dataset is formatted limits the type of inputs that you can use to perform your unsupervised pass. So this is the main problem with this type of existing dataset.

**Interviewer 1:** [00:23:39] Okay, and the model structure for this one and did you..

**Interviewee:** [00:23:44] .We are dealing with networks, I mean with stacks of deep autoencoders.

**Interviewer 1:** [00:23:52] Stacks of deep autoencoders. So, the interaction between these autoencoders, do you remember having any problems because of that?

**Interviewee:** [00:24:02] Probably the main problem is that this type of network cannot be very easily prototyped in Keras. So you have to perform some additional steps, because it is not a common layer. This is a little bit harder to build.

**Interviewer 1:** [00:24:31] So, do you have to do a lot of implementation yourself to make this work?

**Interviewee:** [00:24:37] A little bit more than for the standard layers of Keras.

**Interviewer 1:** [00:24:47] So, when doing these kind of implementations, do you remember or maybe the people you work with complain about some types of bugs that happen too often? Do you remember anything about that?

**Interviewer 2:** [00:25:02] Even small technical bugs, you know, everything.

**Interviewee:** [00:25:06] But you mean related to this problem or to all the problems that we are dealing with?

**Interviewer 1:** [00:25:12] So, the problems you're dealing with that are related to deep or machine learning.

**Interviewee:** [00:25:18] Okay, I mean lately we were dealing with multitasked deep learning. So, one of the problem with that though there are some intuitive generalization of the concept to multitask, like the weighted objective function and the per-task clashing balanced cost-sensitive learning. There are some other parts of Keras framework which do not allow you to handle multitask deep learning, like there is no native implementation of multitask early stopping program.

**Interviewer 1:** [00:26:03] Okay, and what is your approach to dealing with this problem?

**Interviewee:** [00:26:09] We are currently implementing by ourselves.

**Interviewer 1:** [00:26:14] Okay. So the problem here is the limited support by Keras for these types of networks, right?

**Interviewee:** [00:26:23] Yeah, at the moment.

**Interviewer 1:** [00:26:26] Okay. And I would ask the same question, but for this anomaly detection one, do you remember any steps that you have taken that improved their detection of anomalies, because this is a bit different, right? Cause you train only on normal data.

**Interviewee:** [00:26:48] Yeah, yeah.

**Interviewer 1:** [00:26:52] So what is the metric you use? Is it a distance, that how distant is this testing data from what you had on training or? Yeah. What is the metric to classify something as an anomaly?

**Interviewee:** [00:27:11] Usually during the training phase you have only normal data and you minimize the loss function of reconstruction error between the normal data and reconstructed data by the network.

**Interviewer 1:** [00:27:29] Okay.

**Interviewee:** [00:27:30] Then during the test phase, you give the network both normal and attack data, and you are testing whether the distance between the reconstructed data and the anomaly data are exceeding a certain

**Interviewer 1:** [00:27:53] threshold.

Okay. How do you decide on this threshold? Is it like some predefined values that is used across the literature or did you have to do adjustments to get better accuracy?

**Interviewee:** [00:28:06] No no, this is usually a known problem in detection theory, because the threshold, I mean, allows you to increase the detection rate, but also to increase the [inaudible], so there is a trade-off in the increasing or decreasing the threshold. So this is why we also analyzed the performance in terms of receiving the characteristic, which is mainly the performance function of the threshold.

**Interviewer 1:** [00:28:37] The performance function of what, sorry?

**Interviewee:** [00:28:40] The performance is a function of the threshold.

**Interviewer 1:** [00:28:43] As a function of threshold. Okay, okay. So, maybe while we were talking you remembered about any problems or bugs that you have faced?

**Interviewee:** [00:28:59] What, sorry?

**Interviewer 1:** [00:29:01] I was telling that maybe while you were telling us about this project, maybe you remembered about some other problems and bugs that you have faced in your experience.

**Interviewee:** [00:29:14] You mean bugs...

**Interviewer 1:** [00:29:17] All kinds of bugs that you have faced when developing these systems, even the small ones, errors, crashes, anything.

**Interviewee:** [00:29:25] I mean, not significant bugs, the usual bugs that you encounter when you are doing programming in Python, C or whatever other language.

**Interviewer 1:** [00:29:40] Could you name some of them?

**Interviewee:** [00:29:47] Like. A wrongly given type of input, or the inconsistency between the two different layers, or I mean like, ohh, I don't remember any other.

**Interviewer 1:** [00:30:09] Okay. So inconsistency between two different types of layers, is it inconsistency in terms of dimension?

**Interviewee:** [00:30:16] Yeah, yeah.

**Interviewer 1:** [00:30:18] Okay. And wrong type. Do you remember any type that was wrong? And what did you change it to? Is it a an input to the model?

**Interviewee:** [00:30:28] Yeah, exactly. Yeah, it's because usually the input is like a matrix or a tensor. But the first dimension is not aligned to what is the code expecting, so you have to reshape the input, so that the tensor matrix matches the expected input.

**Interviewer 1:** [00:30:54] Okay, thank you. And one of the last questions, do you remember any nasty bugs that you had and that took a lot of your time to fix like, as they usually ask, the most interesting bug that you had so far in these systems.

**Interviewee:** [00:31:10] My experience tells me that usually the bugs that are related to lack of background of how these networks work.

**Interviewer 1:** [00:31:21] Okay. So do you remember any specific one or not?

**Interviewee:** [00:31:27] I mean like when you do prediction and you train a deep learning network, like a recurrent neural network, and yo do not standardize your time series, it won't work because it won't learn anything, for example.

**Interviewer 1:** [00:31:44] Okay. I see. Okay. Thank you. Do you have anything else to add?

**Interviewee:** [00:31:50] No.

**Interviewer 1:** [00:31:53] Okay, good. So, thanks a lot for your time.

**Interviewer 2:** [00:31:58] Thank you.

**Interviewer 1:** [00:31:59] Thank you, it was very nice talking to you. Have a nice day.

**Interviewee:** [00:32:02] Yes, you too, bye-bye.