Episode 13: Conversation with Subutai Ahmad – On Applying HTM Ideas to Deep Learning Networks

Matt:	00:00	Welcome to the Numenta On Intelligence podcast. I'm Matt Taylor, Community Manager, and today I'm talking to Subutai Ahmad, our VP of Research. I've worked with Subutai at Numenta for almost eight years now and I have the utmost respect and admiration for his drive, his thoughtfulness and the calm, studious atmosphere he brings into every situation. It was a pleasure to talk to Subutai about his most recent work, applying the ideas of HTM to constructing deep learning networks. I hope you'll enjoy it. Okay, Subutai, thanks for joining me.
Matt:	00:44	I had some questions for you about what you've been working on in the domain of sparse representations and machine learning.
Subutai:	00:49	Sure. I'm happy to be here.
Matt:	00:51	One of the first questions I always, I've been getting recently on Twitter especially is why are we looking at deep learning? Because that seems to be counter to everything we've talked about for the past 10 years. So why are we looking in the deep learning domain at this point?
Subutai:	01:04	Yeah, this was this was interesting. It's a, it's a change for us this year compared to previous years. So we've been very focused on the neuroscience and understanding neuroscience principles and creating theoretical models of experimental data. And we published a lot there. Sort of early this year. We published the frameworks paper which put a lot of that together into a

		consistent framework and scaffolding that ties a lot of our findings into one place. And I felt at that time it was the right time to start thinking about, okay, we've done all of this neuroscience research, can it actually apply to practical machine intelligence systems and practical systems? And initially the answer wasn't really clear but as we started looking into it more and more, it looked like we can actually. Rather than starting from scratch, we can do a more incremental approach of looking at existing machine learning and machine intelligence systems, start incorporating neuroscience principles and keep expanding and improving on the functionality there. And with deep learning in particular, there are some really big fundamental problems with deep learning despite all of the successes that it has. It has a lot of really key issues that needs to be solved there to get to truly intelligent systems.
Matt:	02:21	Right.
Subutai:	02:21	And the neuroscience work we've done so far I think could really impact the deep learning world and impact practical systems. So that's really the reason I think we, we moved, started moving this way.
Matt:	02:32	So that's an interesting subject to go down. Can you talk about some of these inherent problems in deep learning and what types of things that we see in neuroscience can help in those situations?
Subutai:	02:43	Yeah, and these are - everything I'm going to say is sort of generally acknowledged within the field as well. But there are sort of deep learning has been incredibly successful. But it's nowhere near the capability of human intelligence, which we think of as extremely flexible and dynamic.
Matt:	03:01	Well, voice recognition is really good, but voice understanding is really still very bad.
Subutai:	03:06	Yeah. and so some of these things would be and we've talked about these for a long time, but the idea that a system should be continuously learning. Today's deep learning systems are very static and rigid. You train them and then that's it. They're actually not really learning systems. They're more, they're trained and then they're static.
Matt:	03:26	We used to call them online, they're not online, not continuous I mean people try hacks or ways to, batch train them or get them to train at certain intervals, but that doesn't address the

		underlying problem, which is, it doesn't, it doesn't, like in our brains, our synapses are constantly updating as we learn with every time step. Right?
Subutai:	03:48	Yeah, exactly. And, and in the deep learning context, this is an active area of research and the deep learning world you know, they have this thing called catastrophic forgetting where as you learn new things it's very easy to forget the old stuff unless you really pay special attention to it. But from the neuroscience stuff we've done continuous learning is relatively straightforward. And we've shown this in different isolated settings in the past. If you have really sparse representations and a better neuron model and a predictive learning system you can learn continuously without forgetting the old stuff.
Matt:	04:24	And the neuroscience as far as the synapses and the neuron level, that's been well known for a long time about how those things learn.
Subutai:	04:31	Exactly. Yeah. Yeah. And more and more is being learned, ah, all the time there. Another big thing is robustness. So neural networks are known and deep learning systems are known to be very fragile and sensitive. If you start adding noise or other things, their accuracy drops quite a bit.
Matt:	04:48	Adversarial attacks, perhaps?
Speaker 3:	04:50	Adversarial attacks are sort of the extreme example of that. But we know the human brain is not as sensitive as these networks are. Right. And what we can show is that if, if you have really high dimensional, sparse representations, those networks and also be a lot more stable against noise, then then you know, dense systems.
Matt:	05:11	So let's talk about that. That's one of the topics I wanted to kind of break apart. The idea of high dimensionality is one, cause that's a topic on itself. And then the sparseness within there, it's, I think it's hard for some people to think about dimensionality from a mathematical perspective because we're always living and breathing in three dimensions, you know, through time. But when we talk about high dimensionality and deep learning, or in sort of neural networks, what does that really sort of mean? Like if you take an image, how many dimensions does one image have?
Subutai:	05:42	Yeah. So in the dimensionality, what we're talking about is dimensionality of the vectors that are at play. And so if you

think about a neural network, you have a bunch of neurons or units that are projecting to another set of neurons. And so you have one at any point in time, you have a vector of activity. And if you have 10,000 neurons, then that's a 10,000 dimensional vector, right? And it's feeding into another neuron. Now, this layer that has another, that represented by another big vector, right?

Matt: 06:12 So, so these neurons in well, in our, in our brains and, and deep networks are trying to judge somehow some activity over across this huge amount of space. This huge space.

Subutai: 06:21 That's right. Yeah. So we might be living in a three dimensional world, but the information that's actually in these neural networks and these incredibly high dimensional spaces, and that's what it offers.

Matt: 06:34 And so what, why does sparsity help? I mean, that's really what we're trying to nail down. Why does it help to make a deep network sparse?

Subutai: 06:43 Yeah. So from a robustness standpoint, what we can say is that if you have, what does robustness mean and what is kind of stability mean in this context? You know, if you add noise into the input, you want the output of the layer to be, to not change much, to be pretty stable and insensitive to the noise that's coming in. Right. And what happens with dense networks is if you change something in one place, it affects everything, whereas with sparse networks, because you have mostly zeroes changes in one place tend not to affect you know, representation as much. And so this is a case where if you have a vector of outputs, mostly they're zeros, but you have some non zeros and your connectivity or the weight matrix is also sparse, then changes in the input are going to have very little impact on changes in the output under certain conditions. And that's and the main reason is every, mostly these things are zero. And so most of the time, it's only a small subset of small clusters of points that are really impacting one another.

Matt: 07:52 Is it a stabilizing effect on the whole network? Sort of ?

Subutai: 07:56 Yeah, I think so. As long as you're in the right kind of mathematical regime that's dictated by the equations, then then you get representations that are incredibly stable and incredibly resistant to interference from random stuff going on.

Matt:	08:09	Right. So can we talk about the, the idea of sparse really in the realm of deep learning now we're going to get there in this podcast. So can we talk about the difference between the connections between layers? We can just talk about hidden layers cause it's sorta there's a generic way to apply sparsity right, to connections and then to activations. So can you break that apart?
Subutai:	08:30	Yeah. So there's two sets of vectors really that we're talking about. One is the vector of activations. So at any point in time, what is the output of a given neuron and that, and the output of all the neurons in a layer can be represented as a vector. Then that layer projects to another layer. And if you look at one of the units in the destination layer, there's another vector of weights that indicates what the connection is to each of these input layers. And so when you have an input and you want to find a output of one of these units, you do a vector multiplication or a dot product between the weights and the activations. And that gives you the output. So though, so there's two places you can be sparse. You can be sparse in the activations. So maybe most of these vector, most of these units are zero in the, in the incoming layer or the weights themselves could be sparse. So that means most of the connections are actually zero. Only a small number are connected.
Matt:	09:35	So, okay. So, cause if it's zero, there's nothing to multiply.
Subutai:	09:39	That's right. Right. And if a weight value is zero, it doesn't matter what the input value is for that unit, it's not going to make, have any impact on the, on the output.
Matt:	09:48	And so that's sort of going back to the neuroscience, that's an idea from neuroscience, right? Could you talk about that a bit?
Subutai:	09:55	Yeah. So it's in the neocortex where it's known for a long time that the neocortex is extremely sparse in just about every way you can imagine; there are more ways than these two actually. So the neocortex, the activity of neurons at any point in time is extremely sparse. So as, as I'm talking to you and you are talking to me you know, the neurons in my neocortex are firing, but typically it'll be less than 2% of the neurons at any point in time are firing. And quite often it's significantly less than 1%
Matt:	10:28	For the most part, no matter what's sort of happening in your environment, it's always stable.

Subutai:	10:32	It's always a, well, it's always sparse. It could be moving around, but at any point in time it's really sparse. And so it, that is incredible levels of sparsity it's, you know, 99 more than 99% of the neurons at any point in time are silent. And deep learning systems are not like that.
Matt:	10:49	Right. They're very dense.
Subutai:	10:51	They're very dense. The other side of it is that if you look at the connections and the synapses from one layer to another and if you look at the projections, those are also extremely sparse. Most of the connections that could exist don't exist.
Matt:	11:08	You think about one neuron and all of its dendrites and all of the thousands and thousands of synapses across all those dendrites. If, if this cell body is waiting for some stimulus to respond, that's a huge space, to be observing. Right?
Subutai:	11:21	Yeah, exactly.
Matt:	11:22	Yeah, it makes sense.
Subutai:	11:23	Yeah. And we should get to the neuron units and the dendrites themselves cause that there's, you have other types of sparsity that come into play.
Matt:	11:31	Go for it.
Subutai:	11:31	Okay. So in the neocortex, the, neurons are a lot more complex than in, in deep learning. You alluded to the, you know, dendritic tree, the, the complex set of dendrites and that's where all the inputs to a neuron come into the, onto the dendrites. Right. it turns out those dendrites are very nonlinear and complex. And at any point in time particularly when you get further out from the neuron, the dendrites themselves are tiny, sparse computing devices. So isolated segments of the dendrites are recognizing sparse patterns, and acting on their own, independent of the other parts of the dendrite. So the neuron itself has lots of these tiny, sparse computers spread throughout the dendrites, this is something called active dendrites in the neuroscience literature. Um, and it's, that's pretty interesting. Again, it's very different from how deep learning systems work.
Matt:	12:31	We call it coincidence detectors sometimes.
Subutai:	12:33	Yeah.

Matt:	12:34	But the idea is that those, each one of those little things could send a signal to the cell body to let it know something's about to happen.
Subutai:	12:41	Yes. You know, some part of the dendritic segment could detect a coincidence of other neurons firing and that's a sparse pattern that's coming in and it could then initiate a dendritic spike.
Matt:	12:53	I don't want to go off on a tangent, but how could we apply that idea to deep networks?
Subutai:	12:58	Yeah. So I think that's something we're looking into. I think that's going to be key to doing continuous learning. Cause you, you take two properties together. One is that these sparse vectors are very unlikely to interfere with one another. They're very robust. If you have a random, sparse pattern, it's very unlikely to collide with another random sparse pattern, and so, so you have that. So sparse patterns don't interfere with one another. And then you get to these dendritic trees and each one is independently computing these sparse patterns. Right. Because the neuron is continuously learning, it can learn new patterns dynamically, in different parts of the dendritic tree without affecting the other sparse patterns cause they're independent and mathematically they're unlikely to interfere with one another. Highly unlikely. And so you can have a continuously learning system that avoids this catastrophic forgetting problem. You'd just be adding new synapses and new things to one part of the neuron without effecting the other parts of the neuron.
Matt:	14:06	And potentially the old learning could be applicable to the new space.
Subutai:	14:10	Yeah, exactly. Yeah. If there is a, if there is a close match, you know, it's going to be very applicable cause it's highly unlikely to happen by chance. And so you would learn that, but most of the time you'd be learning other things. Um and this allows you to do hopefully, cause I'm sort of continuous learning really in a really stable way.
Matt:	14:28	It gets rid of the brittleness we talked about in deep learning weights where you change the weights and it could throw everything off.
Subutai:	14:35	That's right. Yeah. Yeah. Cool. so that's one of the key principles through which the brain does continuous learning, I think.

Matt:	14:43	Um great. So is there another type of sparsity or we're going to talk?
Subutai:	14:47	Um yeah, so another type of sparsity would be, so neurons are these independent, sparse computing devices. The learning on a neuron is also very sparse. So in a deep learning system, when you learn, every weight gets updated pretty much. Um but in a neuron only this, the little segment that detected the pattern actually gets updated, right? So the learning happens in a very sparse way on a, on a neuron. And that's critical as well.
Matt:	15:15	Very different than gradient descent.
Subutai:	15:18	Very different. There are a lot of differences there. It's very localized and this again, helps continuous learning cause you, when you are learning a new pattern, you only update one part of the weights and you leave everything else untouched. And it's a tiny percentage of the entire neuron's set of synapses.
Matt:	15:34	Cool. Yeah. So, well, let's talk quickly about the paper. We haven't talked about this in the podcast, although it's been out for awhile on arXiv.
Subutai:	15:45	It's on arXiv now, yeah.
Matt:	15:46	How Could We Be So Dense is the title of the paper? What's the subtitle, I forgot?
Subutai:	15:52	Uh I think I changed it a couple of times. I'm not sure. I think it's like the power of sparse representations or the robustness of sparse representations.
Matt:	15:59	Yeah. Something like that. We'll link it in the show notes. Um but, so let's talk about building actually taking deep learning networks, like typical architectures and how we can convert them into sparse architectures. Yeah. So like a CNN for example, a standard CNN, well, how would you, what's the process of making that a sparse network?
Subutai:	16:21	Yeah, it's actually turns out to be very straightforward. So in the paper what we do is we go through the mathematics of sparse representations and show how it's very stable. And then the second part of the paper, we show how it can be applied to deep learning systems. So in a convolutional network, you have two types of layers. You have the convolutional layers and linear layers. And basically the connection matrix having that be sparse is pretty straightforward. You just randomly initialize a

		whole bunch of those weights to zero and you keep them fixed at zero. So there's like a mask over the weights that maintains
Matt:	17:00	Permanent masks for the whole, the life of the model that's going to be this permanent.
Subutai:	17:04	In this paper, but we did is we just created a single static random mask over at each weight matrix and kept that, those weights to be zero throughout. That's very straight forward.
Matt:	17:15	See that'd blow some people away. You think like, if you did that from the deep learning perspective, you're ruining the network in some way.
Subutai:	17:20	Yeah. But in reality, it turns out that these deep learning systems are often very over-parameterized. They have way more weights than they need. And so you can actually do this and still have a pretty sparse weight matrix and still have it still work. So that gives you a sparse connections. And the way we did sparse activations is very similar to the way we did it with the Spatial Pooler in, in our HTM theory, which is we just look at each layer and select the top K most active cells and keep those active. And the rest are set to zero.
Matt:	17:58	And this is inspired by the idea of mini column structures and yeah, and quartet and neocortex and those being sort of groups with similar proximal dendritic trees.
Subutai:	18:10	Yeah, it, it's, it's, it's closer to this, it's almost identical to the Spatial Pooler where we have a local inhibition um and you know, if a unit is really strongly active, it's going to inhibit its neighbors. Right. And in the, in the neuroscience, in the neocortex, you have these inhibitory cells that that form sort of local inhibitory networks. And we think that's how the sparsity is, is in part created, yeah, enforced in the neocortex. So we have this K winner take all network or system for each layer.
Matt:	18:44	So that's like the activation function.
Subutai:	18:46	That's the activation function.
Matt:	18:47	Instead of like a tanh or a ReLU.
Subutai:	18:50	Yeah. It's actually very similar to a ReLU because in the ReLU anything above a zero is kept active. Here, the threshold point is dynamically determined based on the activity of the other units.

Matt:	19:03	How sparse do you want it? Right.
Subutai:	19:04	Well that's a good question. In the, if we were to match the neuroscience, we want it to be like 98, 99% sparse in the paper we were closer to 80 to 90% sparse. So 10 to 20% non zeros I'd like to get to the point where we're much sparser than that. And then you can do the same thing with the convolutional layers. It's slightly trickier cause there's weight sharing and sub, but it's, you can do the same, same basic idea.
Matt:	19:34	So what does it get you? What do you get from sparsifying networks?
Subutai:	19:37	Yeah, so firstly, so surprisingly so here we have sparse activations and sparse weights, which is extremely rare in the machine learning community. And when I mention this to deep learning people, they're kind of surprised. Like how could that possibly work? It turns out it works really well. So we've shown that for many data sets, the three data sets now that the accuracy level doesn't change so you can get the same level of accuracies that you do with dense networks. Um and, but when you start adding noise into the inputs, the sparse versions are much more robust to random noise into the inputs then the dense versions are.
Matt:	20:20	Which is good cause we've always said that for years we've said sparsity should help with noise.
Subutai:	20:26	Exactly. Yeah. And if, you know, the math is not at all a surprise, it has to be that way. But it was kind, it was kind of nice to see that even in a deep learning scenario, you can maintain that property. Right. and so this just shows that through sparsity if both of these things are sparse, you get representations that are inherently more stable than dense representations in a deep learning system.
Matt:	20:52	Same with the similar accuracies and yeah. And what are the benchmarks we were working, I know MNIST.
Subutai:	20:58	Yes. MNIST is kind of a basic one that you start with whenever you're doing something new. So it works really well with MNIST, then we tried it with CIFAR 10. And so, and then there's we've also tested with audio with the Google speech commands dataset, which is a sort of a data set of one word spoken commands. And if the results hold in all three of those things, a scenario for all three of those benchmarks. And the other nice thing was we tried different network architectures to, so one

		was a simple Lynette style convolutional network. We also did it with VGG 19, which is a much more complex convolutional network, much deeper.
Matt:	21:41	You can apply the sparsity throughout-
Subutai:	21:43	Throughout the network. And it works. And we've also done it with a version of densenet, which we call NotsoDenseNet. So dense nets are, have been used in image net benchmarks and, and larger benchmarks as well. And it works on all three of those different architectures.
Matt:	22:01	Well, it seems like with all these zeroes, eventually we should be able to get some computational gains with that sparse multiplication. Right?
Subutai:	22:11	Yeah, yeah. So you, you know, going back to the question of what is the benefit of sparsity, another big benefit is computational things because if there's a zero, you can ignore that piece. And traditionally it's been very hard to exploit that in machine learning because of GPUS are inherently not as good at handling sparse structures. But the brain is extremely sparse and because of that, it's extremely efficient. I think that in the neocortex, the entire brain like runs on 30 Watts of power or something like that. And this is primarily, I think, due to the extreme levels of sparsity.
Matt:	22:48	It's like a lightbulb. That's crazy, right?
Subutai:	22:50	Yeah. so in theory we should be able to get to extremely sparse structures. And in the paper we lay out some of the just the level of non-zero computations that are going on compared to a dense system and there's at least 30 to a hundred X a difference in the number of non-zero computations. Yeah. The trick will be finding the right hardware architecture and we're starting to explore that. It's still too early to really say definitively anything right now, but there are hardware architectures out there that could exploit sparsity. And if that happens we think we could get tremendous computational benefits as well.
Matt:	23:27	But what about software? Aren't there sparse matrix multiplication software libraries?
Subutai:	23:31	Yeah, I think software libraries, you can get some benefits and we're looking at that as well. But to really run large networks, you need hardware acceleration and it's going to be difficult to run really large scale stuff just on a, on a, purely on software.

Matt:	23:45	Anybody else in the deep learning world focused on sparsity? Is anybody else? Would they want hardware that runs sparse calculations?
Subutai:	23:53	I think so. I think the, there's, there's a bunch of people looking into this already. So I think that's nice to see is that particularly over the last year, I think there's been a resurgence of interest in sparsity so I think our timing is really good. No one has really, as far as I know, really looked into robustness with sparsity and doing both sparse activations and sparse weights seems to be really rare. But yeah, there are definitely a bunch of other labs looking into this as well.
Matt:	24:22	Good. We'll be in good company. So what's the future of research look like for Numenta right now? We're sort of all in, on, on deep learning at the moment. Do you have further plans after you go through sparsity? I know we've talked about continuous learning in the past. What else are you working on?
Subutai:	24:42	Yeah, so I would say we're not so much focused on deep learning as we're focused on practical machine intelligence systems. And currently deep learning is the best example of that. So in terms of our research roadmap, so sparsity is, you want to start with sparsity, say incorporating sparsity everywhere and showing it's robust and showing its fast. Those are the first steps. And then continue adding this notion of active dendrites and a more complex neuron model would allow us to think about continuous learning. And then just as we did with HTM- going to a temporal memory like structure where you have each layer has its own recurrence, a recurring set of connections. What this will allow you to do is again, just like in our old temporal memory, we can not only do continuous learning, but we can do that in an unsupervised manner. So
Matt:	25:33	Recurrence meaning connection to itself, right? So one layer having connections to itself. That's a temporal memory or sequence memory in HTM.
Subutai:	25:41	Exactly. Yeah. And the way we did it in HTM I think could really apply to deep learning systems as well, which is the system is constantly making predictions about what's about to happen. And then when you get the actual data about what happened, that serves as an error signal. And you can immediately do unsupervised learning on that and that can be done in a continuous learning setup because you're using sparse representations in these active dendrites. So now all of a sudden you have something that doesn't require as much label

		training data can really deal with the streaming sensory inputs. And is constantly learning through these predictions.
Matt:	26:20	Would we still be having to apply gradient descent over top of all this as we go?
Subutai:	26:25	Yeah, that's a great question. You know, in the brain we don't have a strict backpropagation-like structure or learning system. So over time, as our systems become more and more, you know, critical, I think the, the need for backpropagation will lessen, so to do this predictive learning if it's happening at every layer independently, you don't need to do backpropagation as much there. And we might still for a while have gradients flying through just because currently there's no better way to create really scalable systems. But over time as we learn, as we incorporate more and more principles from the brain in there, hopefully we can, we can get rid of that too.
Matt:	27:08	That sounds exciting.
Subutai:	27:09	That's when it's not deep learning anymore.
Matt:	27:10	Yeah, I guess not.
Subutai:	27:11	Yeah, exactly.
Matt:	27:13	It's still neural networks.
Subutai:	27:14	It's still neural networks and that's fine. I mean neural that works are supposed to model the brain because that was the whole reason there came into being, so, yeah.
Matt:	27:22	Yeah. Well that's great. Subutai. Anything else you want to talk about while you have this opportunity?
Subutai:	27:29	No, I think it's, I think from the external world, you know, you kind of said, it seems like, Oh, why are we jumping into the deep learning bandwagon? That's really not what we're doing. I think we feel very happy about where we are on the neuroscience and now we're going and picking off all of those pieces, everything we've done there and start to implement them in practical systems. And I think the timing is right for that and it's I'm really excited about it because a lot of potential there.
Matt:	27:55	Well, thanks for your time, Subutai. We're doing fist bumps now. Take care.

Subutai:	28:00	All right, take care.
Matt:	28:01	Thanks for watching.
Subutai:	28:02	Bye.
Matt:	28:09	Thanks for listening to the Numenta On Intelligence podcast. My name is Matt Taylor. I am the community manager and engineer for Numenta. My guest today was our VP of Research Subutai Ahmad. If you liked this content, you should also check out our YouTube channel. I've been live streaming our research meetings and journal clubs every week.