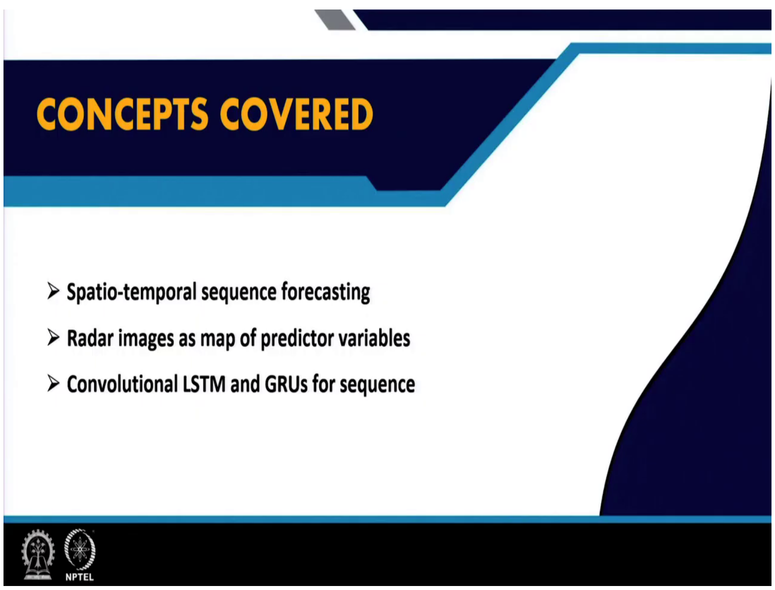
Machine Learning for Earth System Sciences
Prof. Adway Mitra
Department of Computer Science and Engineering
Centre of Excellence in Artificial Intelligence
Indian Institute of Technology, Kharagpur

Module - 04
Machine Learning for Earth Observation Systems
Lecture - 32
Precipitation Nowcasting from Remote Sensing

Hello, everyone. Welcome to lecture 32 of this course on Machine Learning for Earth System Sciences. We are in module 4, where we are discussing like how Machine Learning can be used for Earth Observation Systems and in the topic of this lecture is Precipitation Nowcasting from Remote Sensing.

(Refer Slide Time: 00:43)



So, nowcasting is a concept which we have discussed briefly earlier also for extreme weather nowcasting and so on. So, here the like we will the input sequence will be some kind of a spatio-temporal like a spatio-temporal sequence and the input is that and the output which we are trying to forecast is also a spatio-temporal sequence.

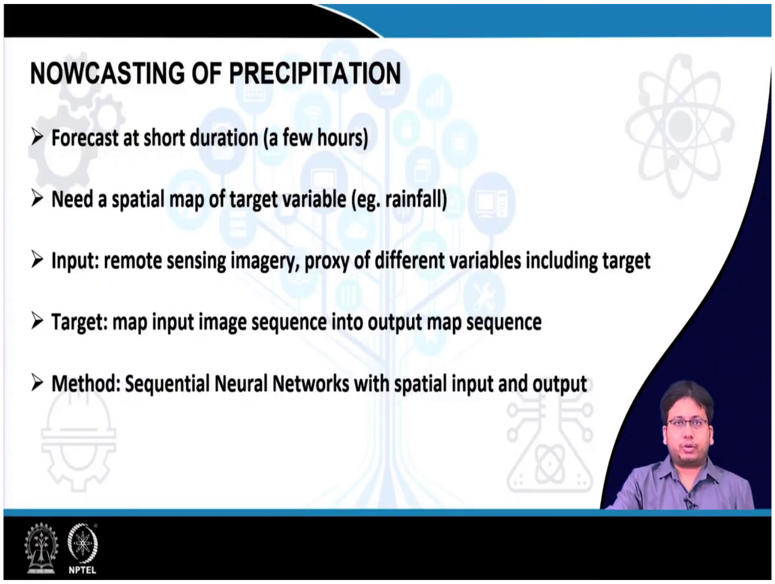
The however, in this case the predictor variables like they will be presented as some kind of like as a like instead of having actual measurements of these predictor variables, we will be having maps obtained from the remote sensing.

And as we have discussed, in the just the previous lecture that these remote sensing like they can act as proxy for different these of these meteorological variables. They may not be directly measured by the remote sensing imagery, but the remote sensing can imagery can act as a proxy for all those variables. So, in this work we will like in like in this lecture the where research papers that where we are going to discuss, they do not specifically measure the those variables rather they use the like the remote sensing images.

Especially, the radar images like directly on which they apply the machine learning algorithm. The covariate information we can say we can like look at it in this way that covariate information is implicit in the form of some proxies in the neural network. But, they are never brought out explicitly, but rather it is directly fed into the machine learning or deep learning model which brings out like which maps the those remote sensing imagery to the quantity of interest which in this case is rainfall and so, like we will see different models machine learning models for this purpose.

We will see convolutional LSTM which we have discussed like briefly in earlier also and we will also see GRUs or gated recurrent units for the sequence modelling.

(Refer Slide Time: 02:53)



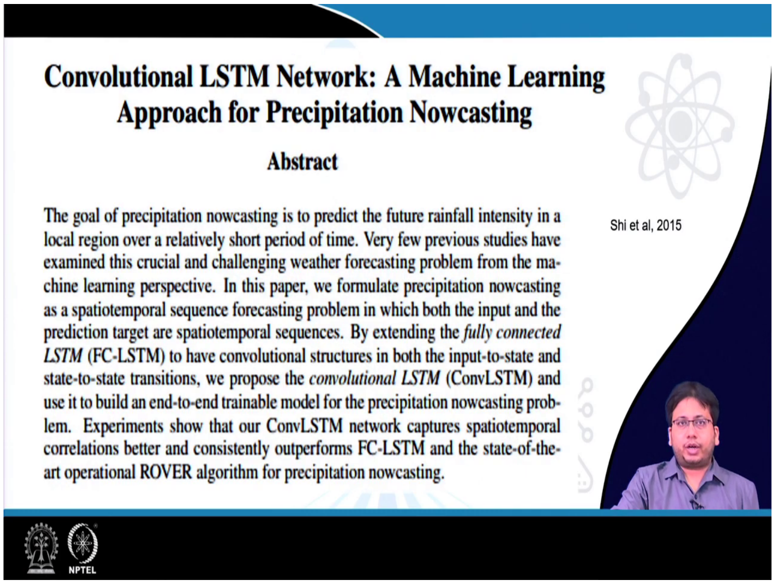
So, now coming to nowcasting of precipitation as has been already discussed so, this is basically doing the forecasting at a short duration typically a few hours. And however, in this likes in some unlike some of the other nowcasting applications which may we may have considered in this case we need a spatial map of the target variable that is at every location I should be able to make an estimate of how much rainfall is going to happen after say 5 hours or something like that.

So, the input in this case is a remote sensing imagery which acts as a proxy of different a variables including may be the target variable that is the rainfall. So, like when we are obtain like obtaining the a radar imagery at this point of time like the pixel intensities and for corresponding to different locations, they may have different they may be encoding different information including the how much precipitation is happening right now. And the aim is based on this to construct the precipitation map for the maybe like 5 hours in advance and so on.

So, the target is to develop like a map of the input like is to develop something like a map sequence of the target variable in this case rainfall. So, like we what we are trying to do is to build a mapping of like you can say a mapping of maps the input is like the radar the radar imagery which is like a proxy for map of different variables and the output is also going to be a map of the target variable rainfall and not only that the input is a sequence of maps, the output is also going to be a sequence of maps.

And, for this purpose we are going to consider different kinds of sequential neural networks which have the spatial like which are suitable for spatial input and output.

(Refer Slide Time: 04:50)

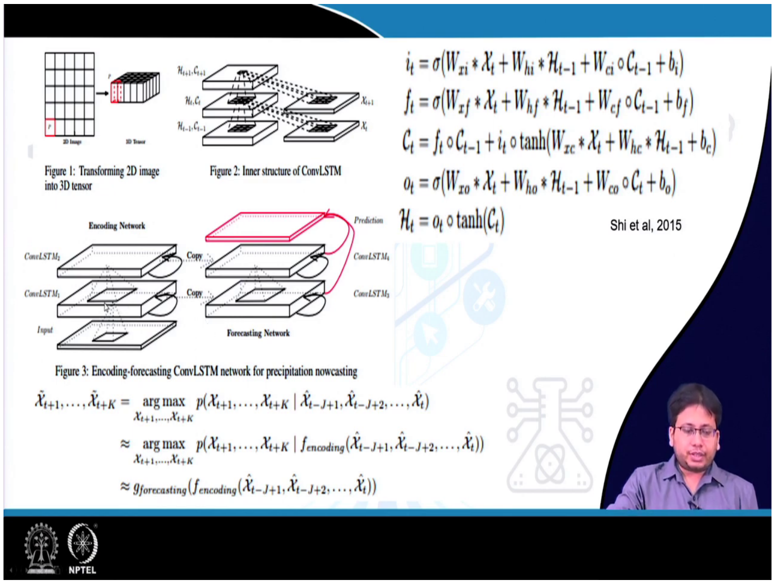


So, the first model which we are going to consider for this purpose is ConvLSTM which we have mentioned a couple of times in our previous lectures also. The goal of precipitation nowcasting is to predict the future rainfall intensity in a local region over a relatively short period of time. A few previous studies have examined this crucial and challenging weather forecasting problem from the machine learning perspective.

In this paper, we formulate precipitation nowcasting as a spatiotemporal sequence forecasting problem in which both the input and the prediction target are spatio-temporal sequences. By extending the fully connected LSTM to have convolutional structures in both the input-to-state and state-to-state transitions we propose the ConvLSTM and use it to build an end-to-end trainable model for the precipitation nowcasting problem.

Experiments show that our ConvLSTM network captures spatiotemporal correlations better and consistently outperforms FC-LSTM and the state-of-the-art operation operational ROVER algorithm for precipitation nowcasting.

(Refer Slide Time: 06:05)



So, this is what the ConvLSTM looks like this I think we have these figures we may have seen earlier also. So, first of all the this is the standard LSTM like a like as mentioned earlier. So, I will it is it has a this hidden state and it has the cell state like a which represent the long term and the I mean the short term and long term memories respectively I like.

So, a like so, a based on the input X_t at every given state like the H_t the short term memory is updated and the cell state can also be updated. So, there is like internally a signal called forget is generated based on the input and the current memory storage. So, if the input; if the forget input is activated, then the cell state is reset that is the long term information that is present gets erased and new information gets stored in it for a preserving for the next steps and so on.

And, these H_t the hidden variables these contain the short term memory I mean the short term memory which is updated at every given step as the input comes in and like we can have the output also at every given step. Now, to so, that is what a convolutional unit looks sorry an LSTM unit looks like. Now, to increase the representative power to and have a longer long term memory and so on, it is like we often use a stacked LSTM unit that is instead of having a single LSTM unit we just connect several LSTM units like in parallel and so on.

And, then so, here is the input is now in this case the case the input is not a ordinary sequence, but a sequence of like we can as we can say a spatial map which is effectively an image. So, and we know that whenever the input has some kind of a special structure as a I mean a 2D or 3D spatial structure as happens in an image. Then, the full the normal fully connected approach which is like which can be represented in a neural network by this like this weight multiplications and so on.

But this does not work very suitably, rather convolution is a more useful feature is a more useful approach because it is capable of a representing the repetitive nature which is present in an image. So, like in the in this case the what is what happens is that this matrix multiplication operations and so on which are present in an usual LSTM, these are actually replaced by convolution operation.

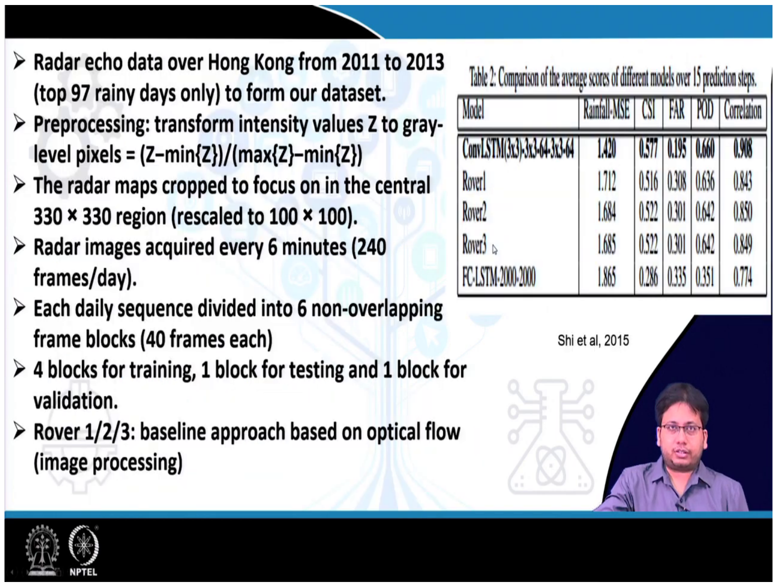
And, the so, like the input this X instead of being a vector or a matrix now becomes like a spatial map on which this kind of convolution operation takes place and these $w's$ instead of these being ordinary weight matrices, these are now turned into convolutional kernels. So, in the way it the whole thing works is as follows. We have an input sequence, input sequence is like is fed to this convolutional LSTM one step at a time at I mean.

Because it is a like a because it is a sequence of images at every point of time one image or one spatial map is fed into the LSTM and the LSTM at each step like it creates some kind of an intermediate representation by passing it through the all the stacked layers and in the different like in the successive steps this it is updated also I mean both the long term as well as the short term memories.

And, like now finally, when the entire input is fed into the LSTM it generates something like an intermediate representation. Now, that intermediate representation is then used for the forecasting purpose that is there is a decoder network which creates the output sequence from that intermediate representation. So, like there is a an encoding step and a decoding or the forecasting step.

So, the this encoding is one set of convolutional LSTM and this output the g forecasting which they have like as mentioned in this paper that is also like a ConvLSTM stacked ConvLSTM. The reason why both input I like the I mean the output also needs to be a ConvLSTM is because the output is also going to be a spatial map or an image just like the input.

(Refer Slide Time: 10:55)



So, the data set they have used for this study is a like radar echo data over Hong Kong from 2011 to 2013. So, this is a 2 year period, but within that they are focused on only 97 rainy days because that is when the observations were available. So, the radar of course, measures it is like a it is the reflectivity it measures. So, the from the radar images, the intensity values first of all the intensity values are like normalized using this kind of an operation.

So, and then like projected to the range 0 to 255 as happens in on an ordinary image. Also the these radar maps are cropped so that they focus on one particular on the particular region of their interest. Now, these radar maps are acquired every 6 minutes that is where we get 240 frames every day. And, now for the for the purpose of training the network what have what they have done is each of these daily sequence of these 2 for 240 frames like so, they divide into 6 non overlapping blocks.

So, if 240 frames are divided into 6 blocks so, there will obviously be 40 frames in each block. Now, for every day we get 6 blocks like this, 4 of these blocks are used for training, 1 for validation and 1 for testing. And so, like this is the ConvLSTM model we with, so they use this for the rainfall prediction purposes as mentioned. So, they like whenever there is nowcasting, then we have this standard indices for nowcasting which we have discussed earlier also for the lightning nowcasting case.

So, there is the POD or the probability of detection, the FAR falls along rate, the correlation coefficient is of course, there and so on. So, they compare the proposed ConvLSTM architecture with a fully connected LSTM in which the input is not provided a in the form of a this kind of a special map, but is simply factorized and then there are the these various rover algorithms.

These rover like this is simply based on the notion of optical flow. So, this is optical flow is a concept in computer vision or machine learning like basically it is used for something like as video prediction. So, suppose you have a video is nothing but a sequence of images in which across which there is some motion, say the video of a car running or something like that.

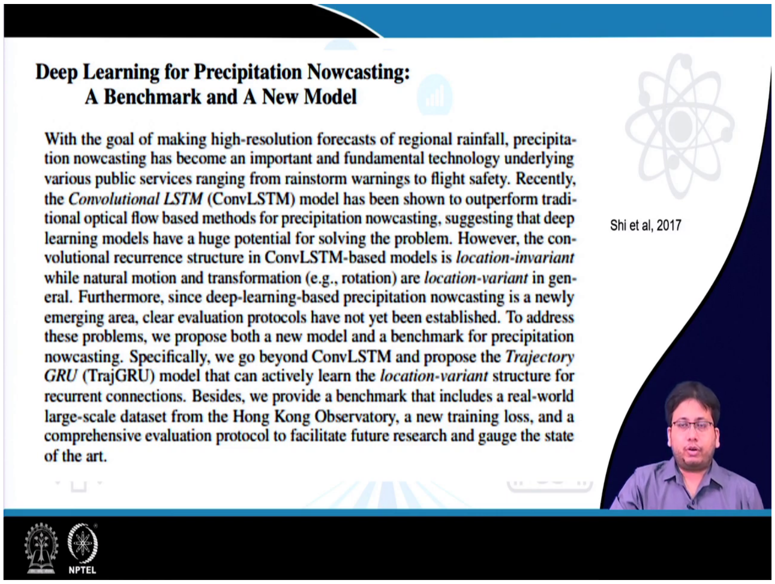
So, between the frames there is going to be some kind some small motion like the objects will have shifted a little bit from one side to another and so on. So, like basically what optical flow does is like at every pixel it measures the shift that may have happened from the previous frame to till the current frame and that same shift it tries to extrapolate to predict what the next frames are going to be.

So, these there so, based on this concept of optical flow there are some rover algorithms which are used for these radar sequence mapping. The that is that is they imagine the radar image sequence as something like a video and based on that they try to like predict the future radar sequences.

Now, these the from the radar sequences there are there is some Marshall's algorithm by which it is actually possible to map the radar inputs to rainfall. So, that is some kind of a parameterized parametric relationship that some meteorologists have come up with. So, using that, they try to predict the rainfall.

So, a so, it is a two-step process – first you like forecast the coming the future radar maps and based on that you like by using the parametric formula you a like calculate how much rainfall can happen in the coming like coming points of times. So, they have compared these approaches and they find that ConvLSTM performs the best it not only over the rovers, but also over the fully connected LSTM which a like emphasizes the fact that a using this convolutional operation here is very crucial.

(Refer Slide Time: 15:15)



Now, building on thus this ConvLSTM model a similar another paper appeared by the same group of authors Shi et al., the like they propose another model which is which they show to be even better than ConvLSTM. With the goal of making high-resolution forecasts of regional rainfall, precipitation nowcasting has become an important and fundamental technology underlying various public services ranging from rainstorm warnings to flight safety.

Recently, the ConvLSTM model has been shown to outperform traditional optical flow based methods for precipitation nowcasting, suggesting that deep learning models have a huge potential for solving the problem. However, the convolutional recurrence structure in ConvLSTM-based models is location-invariant while natural motion and transformation such as a rotation are location-variant in general the that is to say from one like that is when we are considering a convolutional operation.

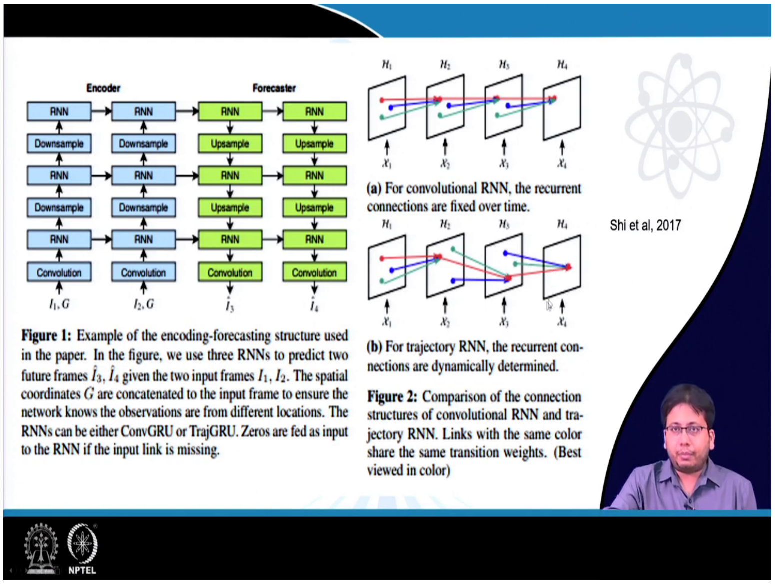
So, it is like the same convolution kernel is used all over the image and so on. But what they are saying is that the like if instead of doing a doing that like if we could somehow differentiate between different the different locations how the convolution will take place or how the like basically how the recurrence structure will take place in different parts of the input they like that might be useful.

So, that is here basically they are trying to focus on the nature of the trajectory that thread over will I mean the trajectory means like at the path take over time taken at different locations. So, that might be different from one location to another and it is this trajectory that they are trying to somehow focus on and model it in this work. Furthermore, since deep-learning-based precipitation nowcasting is a newly emerging area, a clear evaluation protocols have not been established.

To a address these problems we propose both a new model and a benchmark for precipitation nowcasting. Specifically, we go beyond convolutional LSTM and propose Trajectory GRU or TrajGRU model that can actively learn the location-variant structure for recurrent connections. Besides, we provide a benchmark that includes real-world large-scale dataset from the Hong Kong Observatory, a new training loss, and a comprehensive evaluation protocol to facilitate future research and gauge the state of the art.

So, like in this work of course, like in this lecture we will focus only on the TrajGRU model, but this paper also provides a benchmark data set and a like and a way to evaluate this kind of like a radar based or satellite based forecasting of rainfall. We will not go into that, but I leave it as a homework for you to work on.

(Refer Slide Time: 18:17)



So, this is what the general structure the general structure is more or less unchanged compared to the like the ConvLSTM. So, the like at any given point of time the inputs are presented in the form of the spatial maps or the radar images in this case, so the like it undergoes a sequence of convolutions and RNNs and so on. So, like we see the multiple with the RNNs these might be LSTMs or I mean conv like LSTMs or other sequential units which are used in machine learning such as a GRU, graded recurrent units.

So, I will explain to you in a bit what this GRU is all about, but like as you can see that there are multiple stacks of RNNs. So, like first the image undergoes one round of convolution then there is a RNN that is basically there is this hidden variable which is updated. Then there is another round of down sampling followed by again an RNN, this is the second level of the hidden variable and there is further down sampling and again RNN and so on.

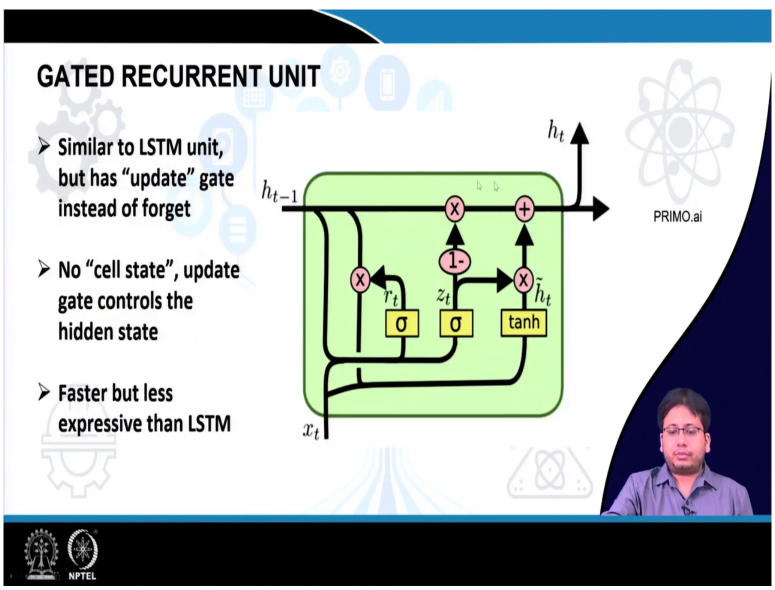
And, these are like the hidden variables these are updated and passed on to the next step where again a new input comes in and undergoes all the convolution and down scaling steps. And, at they are further used to update the memory through the RNNs and so on. So, this way the whole encoding process goes on the in the entire like input sequence is encoded just as already discussed and then there is the forecaster it does the forecasting.

Now, these RNNs can they can be LSTMs they can be or they can be GRUs and so on. So, now, in the convolution as they are saying in the convolutional RNNs the recurrent connections are fixed over times, but they are try like in what they are proposing right now is the trajectory RNN where the recurrent connections are dynamically determined.

That is it is not like when in the ConvLSTM it is like the same thing which is being used all the time, but in like what they are calling it as TrajGRU like as you can see these connection structure this has changed.

And like this is like this changes from one input point like at t equal to 1 it is something, at t equal to 2 it will be something else and so on. And, these the changes which they are talking about this will not be like it is not that these changes are going to be random, but it will depend on the current state I mean the current memory state and not current so on.

(Refer Slide Time: 20:58)



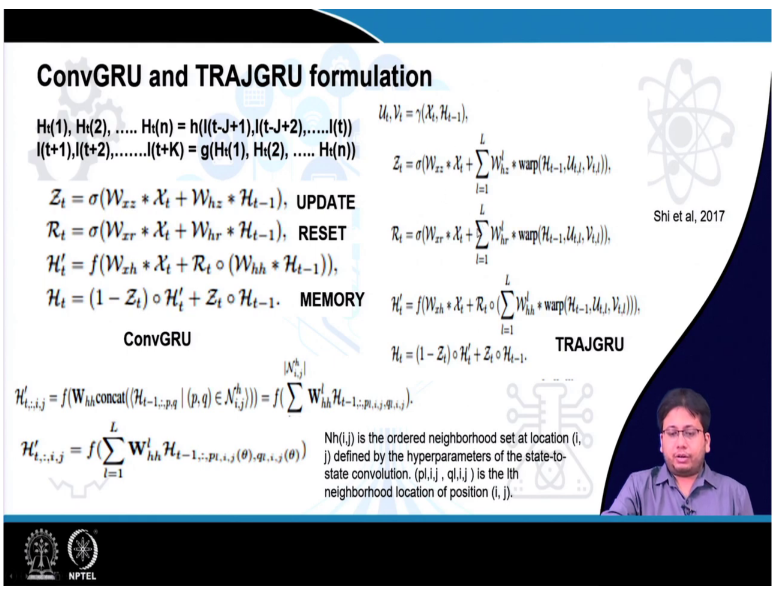
So, and now that brings us to the concept of GRU it is a gated recurrent unit like you can say it is a small smaller scale version of an LSTM, but like it has; like it has slightly less number of operations as an LSTM. So, like it does not a have a like an LSTM has this cell state which is the long term memory and the h the usual hidden state which is the short term memory which is also used in RNNs.

At every point of time there is a forget signal is generated in the LSTM which indicates whether the cell state has to be reset or not. But, in case of GRUs there is no separate cell state nor is there any forget gate instead there is an update signal which indicate that the memory by how much it should be updated that is the if the memory now the single memory now acts as part of like it is a combination of long term and short term memories.

The thing is it is neither like fully updated like it is not fully updated that happens in case of RNN nor like in a LSTM, but rather the update signal indicates how much of it has to be updated and how much of it needs to be like preserved for the coming steps. So, like, this is slightly less expressive as suggested like in case of LSTM the in that the two things the long term and short term are clearly delineated. So, that it will be more expressive in this case it is not that.

So, a this the short term memory itself is trying to you can say accommodate a part of the long term memory. So, it will be less expressive, but it is faster.

(Refer Slide Time: 22:48)



So, this is the what the GRU. So, the like based on that they have proposed two variations one is the ConvGRU and the TrajGRU. So, the in case of the ConvGRU this is like; this is just like the usual GRU where the these stars are the all the operations. Now, in a normal GRU these operations are the usual matrix multiplications and so on where these W_{xz} etcetera these are like you can say that like these are full weight matrices indicating a fully connected neural network layers.

And, so, like this so, first and update signal is generated or reset signal is generated account and the new memory value H is generated based on the current input as well as the past memory and the reset. So, like this reset indicates whether the past memory that should be like should be forgotten or not and also the Z_t is the update signal which we mentioned earlier.

So, accordingly the this like that is if this Z_t is this Z_t this will be a binary variable either variable either 1 or 0 and like so, if $Z_t = 1$; that means, the memory will be will not be updated that is basically the current value of the memory with will continue to be present. But, if $Z_t = 0$ that means, the like whatever new information has been created that information will now be presented will be stored in the memory and the old information will be deleted. So, this is how the ConvGRU works.

So, the ConvGRU simply replaces these fully connected layers by a convolutional layer. So, these matrix multiplications are replaced by the convolution operation and these W_{xz} W_{hz} are these matrices. These basically now get replaced by convolutional filters. So, that is the ConvGRU. So, this is ConvGRU we can as you can see is a like a direct analogy of ConvLSTM.

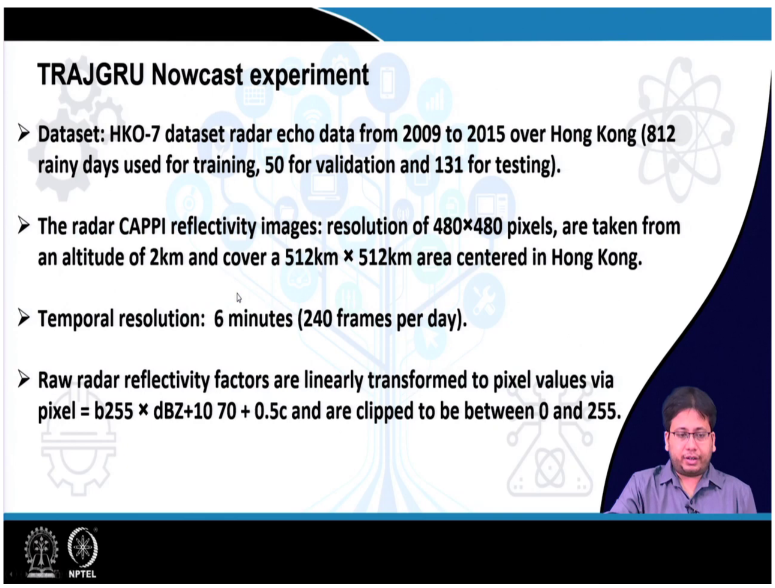
The alternative is to have TrajGRU in which as they have mentioned earlier also like all the these recurrent corrections they like vary from every point of time. So, that is achieved by these two intermediate variables U_t and V_t. So, based on the current input and the past memory as a function of that these two signals U_t and V_t are generated and these U_t and V_t these are used to warp the past memory the H_{t-1}.

So, like this warping of the H_{t-1} this basically helps to change the recurrent structures as they are saying. Now, this warping does not happen like randomly, but as a like a there is this function gamma of the a like which takes the X_t and a the I mean the present input and the past memory and it does some kind of operation on it.

And, now like in case of ConvLSTM this gamma this like it is some kind of a constant function, but in the TrajGRU this a like it is not constant it like it take it some like it is a function that changes it is value and this. So, like basically the like if you look at the H' which is the new information generated at every point of time it is a so, like it takes into account the neighbouring structure of every i, j if the that is they. So, this H is now something like a spatial map where like there are these locations.

So, these, so every location has its neighbouring location. So, like all these parameters of the neighbouring locations are warped as this operation indicates and based on that the new information is generated.

(Refer Slide Time: 26:55)

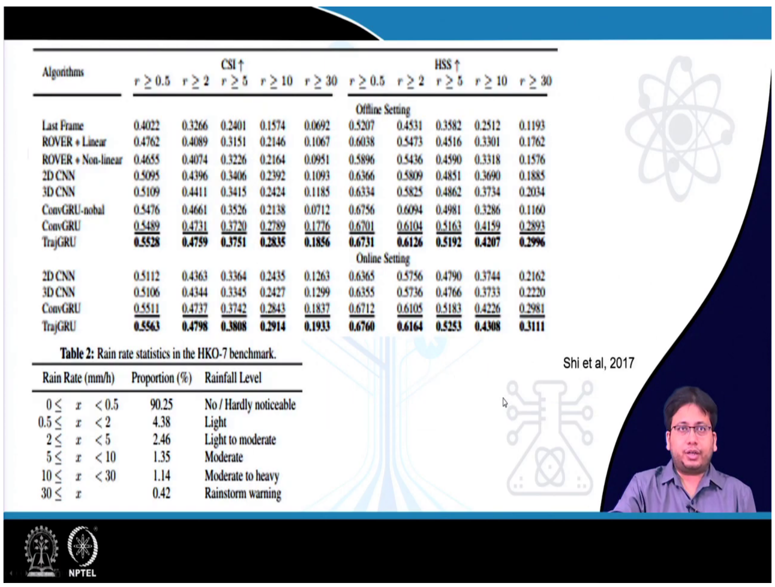


So, now, coming to the TrajGRU nowcasting experiment so, the data set like it is an up scaled version of the previous radar echo data set that was used for ConvLSTM. So, the this is used for the period 2009 to 2015 over Hong Kong and now, there are as you can see if a this has been scaled up. So, instead of 97 there are now 812 rainy days which are available for training, 50 for validations and 130 for scaling. So, these are the radar reflectivity images. Their resolution is $480 \, X \, 48$.

They are taken at an altitude of 2 kilometres and they cover this kind of a this $512\ X\ 512$ kilometre areas over the region of Hong Kong. So, at every kilometre wise like it is the aim is to like make the forecasting. And, so, that these are all they are there is a pre-processing step also where the raw these raw radar reflectivity images they are transformed to the pixel values by according to a formula like this and they are also clipped to the region 0 to 255.

So, that it becomes a effectively an RGB image in the sorry a grayscale image in the visible spectrum on which all these convolution etcetera operations can be done easily.

(Refer Slide Time: 28:09)



Algorithms	CSI ↑ $r \geq 0.5$	$r \geq 2$	$r \geq 5$	$r \geq 10$	$r \geq 30$	HSS ↑ $r \geq 0.5$	$r \geq 2$	$r \geq 5$	$r \geq 10$	$r \geq 30$
						Offline Setting				
Last Frame	0.4022	0.3266	0.2401	0.1574	0.0692	0.5207	0.4531	0.3582	0.2512	0.1193
ROVER + Linear	0.4762	0.4089	0.3151	0.2146	0.1067	0.6038	0.5473	0.4516	0.3301	0.1762
ROVER + Non-linear	0.4655	0.4074	0.3226	0.2164	0.0951	0.5896	0.5436	0.4590	0.3318	0.1576
2D CNN	0.5095	0.4396	0.3406	0.2392	0.1093	0.6366	0.5809	0.4851	0.3690	0.1885
3D CNN	0.5109	0.4411	0.3415	0.2424	0.1185	0.6334	0.5825	0.4862	0.3734	0.2034
ConvGRU-nobal	0.5476	0.4661	0.3526	0.2138	0.0712	0.6756	0.6094	0.4981	0.3286	0.1160
ConvGRU	0.5489	0.4731	0.3720	0.2789	0.1776	0.6701	0.6104	0.5163	0.4159	0.2893
TrajGRU	**0.5528**	**0.4759**	**0.3751**	**0.2835**	**0.1856**	**0.6731**	**0.6126**	**0.5192**	**0.4207**	**0.2996**
						Online Setting				
2D CNN	0.5112	0.4363	0.3364	0.2435	0.1263	0.6365	0.5756	0.4790	0.3744	0.2162
3D CNN	0.5106	0.4344	0.3345	0.2427	0.1299	0.6355	0.5736	0.4766	0.3733	0.2220
ConvGRU	0.5511	0.4737	0.3742	0.2843	0.1837	0.6712	0.6105	0.5183	0.4226	0.2981
TrajGRU	**0.5563**	**0.4798**	**0.3808**	**0.2914**	**0.1933**	**0.6760**	**0.6164**	**0.5253**	**0.4308**	**0.3111**

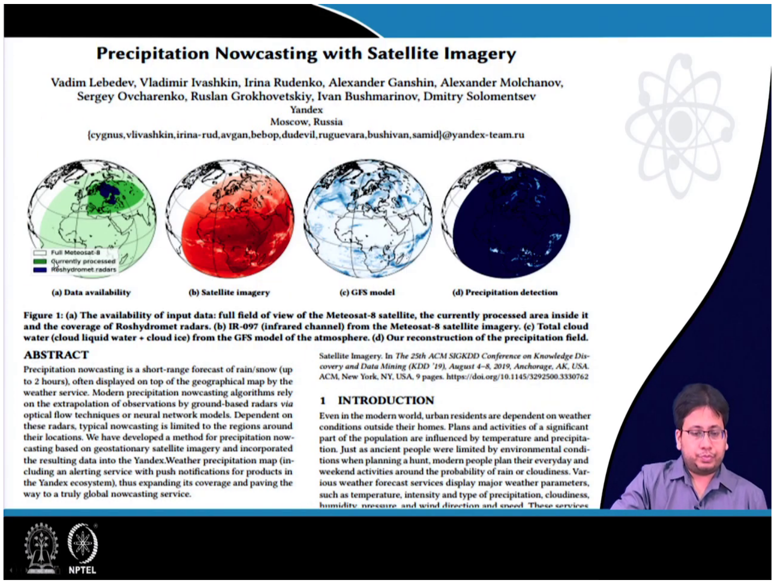
Table 2: Rain rate statistics in the HKO-7 benchmark.

Rain Rate (mm/h)	Proportion (%)	Rainfall Level
$0 \leq x < 0.5$	90.25	No / Hardly noticeable
$0.5 \leq x < 2$	4.38	Light
$2 \leq x < 5$	2.46	Light to moderate
$5 \leq x < 10$	1.35	Moderate
$10 \leq x < 30$	1.14	Moderate to heavy
$30 \leq x$	0.42	Rainstorm warning

And, so, these are some of the results. So, like they be instead of forecasting the exact amount of rainfall they basically do some kind of binning. The rain rate is like is divided into several bins like this and at every bin they try like bin wise they try to make the prediction. And they show that for almost all the bins like you say the bins means like very little or hardly noticeable rainfall light rainfall or light to moderate rainfall.

So, like when they are forecasting the rainfall, rather than they are they effectively forecast that these are the bins of rainfall that are going to happen after say 4 hours or something like that. And, they show that for all the bins the TrajGRU is giving a better result compared to the all the existing methods including like ConvLSTM as well as ConvGRU, ok.

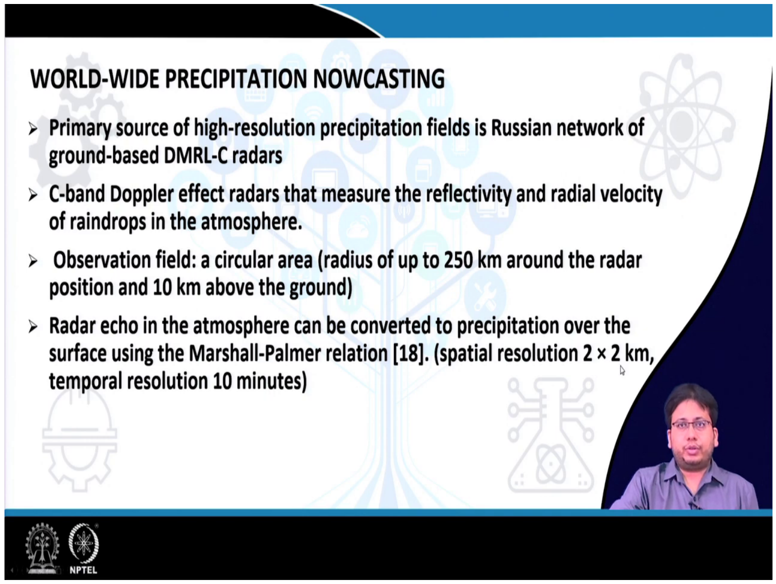
(Refer Slide Time: 29:07)



So, the now we will just briefly discuss one more paper. Here the target is precipitation nowcasting with satellite from satellite or in fact, radar imagery, but here the a aim is rather than focusing on a fixed region let us like Hong Kong, they are trying to build a worldwide map of a precipitation in the like based on the satellite and radar imagery. Now, it might happen that in some regions the radar network is strong so that they have a better data in other places they may not have so much data.

So, like in these, so especially, in this work they have a good radar a network in this entire Russian region which provides them a good source of data. In the other parts of the world they have to depend on the satellite imagery to make the predictions. And, along with it there they also use the simulations from a numerical weather prediction model which is the GFS and using that they make the like the estimation of the precipitation over the all over the world.

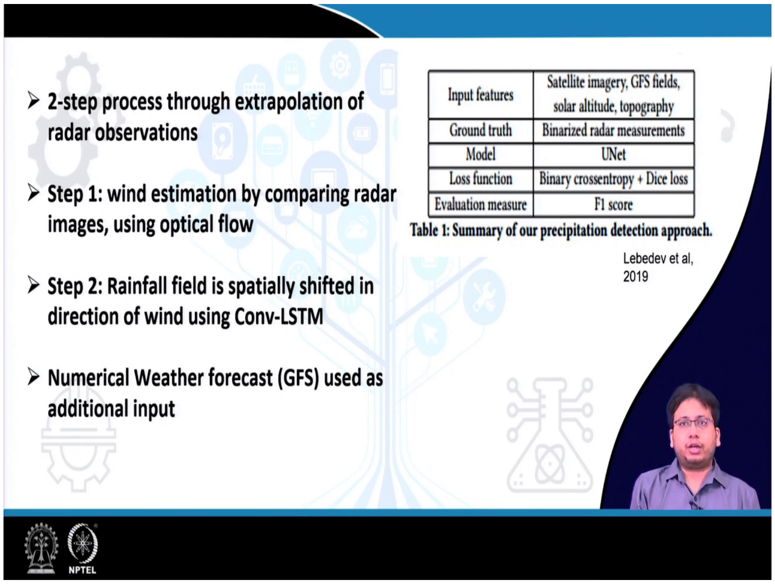
(Refer Slide Time: 30:16)



So, in this case the primary source is of information is the of high resolution precipitation is obtained from the rush network of the ground bed the ground based radars the. So, it is basically C-band Doppler effect radars that measure the reflectivity and radial velocity of raindrops in the atmosphere. The observation field in this case is a circular area of a radius of up to 250 kilometres around a like around the radars position.

So, the echo which is obtained by the radar which is like basically with the measurements of the radar they can be converted to the precipitation, like in the previous case also I had mentioned this that there is a called parametric formula called the Marshall formula which is useful for like mapping the a the radars value with the radar observations to the rainfall values. So, like and it is like this rainfall values are obtained at very high spatial resolution as you can see 2 kilometres per by 2 kilometres at a temporal resolution of 10 minutes.

(Refer Slide Time: 31:26)

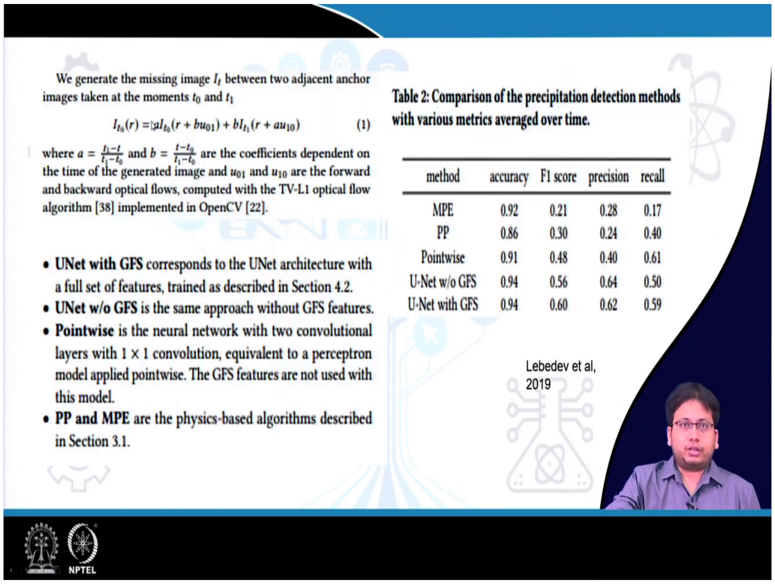


So, like this is basically a 2-step process which they have followed in this paper. So, the 1st step is the extrapolation of the radar observation. So, like this is done using the optical flow which that also we mentioned in contact in the context of the rover algorithm earlier and the step 2 is the rainfall field in the spatially shifted direction of wind a using the ConvLSTM.

So, first of all there is a way they estimate the wind, then they use the common sense that from t a the current time to the next time where they are going to make the prediction by like the rainfall is likely to have shifted in the direction of the wind. And this wind direction is of course, different in different parts of the world because in this case they are making a worldwide forecast of rainfall. So, this is obtained.

So, basically they get the wind map wind direction map all over the world using the optical flow, then the rain the current rainfall map that is shifted according to the rain like according to the rainfall in the next map and the and for this shifting purpose they use the take the help of a ConvLSTM. And, also the numerical weather forecast using a model known as GFS, so, like we have already discussed about the numerical weather prediction models in contact in the context of the light net so, these models are also used as additional inputs.

(Refer Slide Time: 32:56)

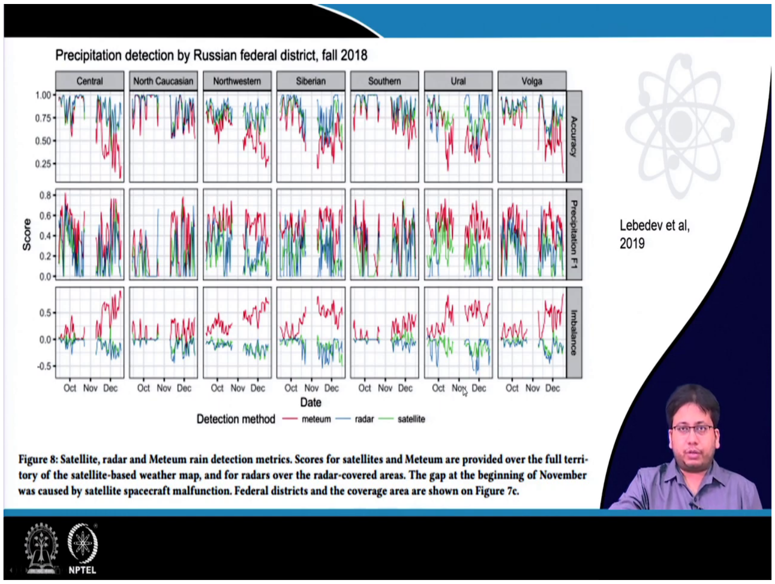


And, so, this is how the like this is like the model they have developed is the is effectively the UNet which we have discussed several times earlier. And so, like they have the UNet which may or may not be accompanied by the input from the GFS which is the numerical weather prediction model. Apart from that like they have compared it with various base lines. So, PP and MPE are some physics based algorithms. So, these are like purely like you can say numerical weather models which are not based on machine learning.

And, then there is the point wise approach where there is a neural network with two convolutional layers it is like, but applied to every location individually. So, like a without taking the neighbourhood effects which is done in the case of like a model like UNet and so on. Also in case of the UNet architecture. So, like if the convolution in this it is used as the convolution operations they are used to take the special effects into account and so on and in the like as these results suggest that this UNet with GFS that often performs the best.

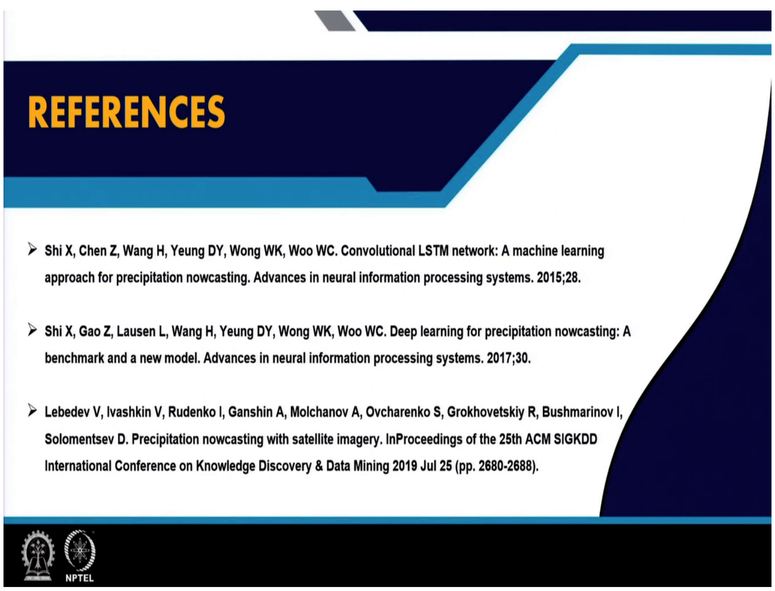
So, in UNet without GFS can also do well in certain situations, but in general the UNet the proposed UNet model along with the numerical weather prediction that gives the best results.

(Refer Slide Time: 34:31)



And, these are like some of the results in the in different period in different parts of the world because they have done the worldwide estimation and they have also done it in different months. So, in different months also they have plotted the results and so on.

(Refer Slide Time: 34:47)



So, these are the three papers that we discussed and that brings us to the end of this lecture. So, we the basic points to remember is that the like when we are trying to estimate the spatial field of a particular variable, we are trying to predict a spatial map of I mean we focus on precipitation, but it can be done for other variables also potentially. So, like in that case it is important to be incorporate a convolutional component into your machine learning model.

And, and if you are if you want to do the sequential prediction you also need to have a sequential model be that LSTM or GRU or something like that. And, it is possible to and you, but also the input can be for the input you need not have actual measurements of the different variables, but rather the radar or satellite measurements themselves can provide as suitable inputs for the a the neural networks have sufficient representative power that will be able to map the radar inputs to produce like a map of the variable which is of your interest.

So, that brings us to the end of this lecture. In the next lecture 2 we will discuss some like some aspects of how machine learning can be used in remote sensing and earth observations.

So, till then, bye.