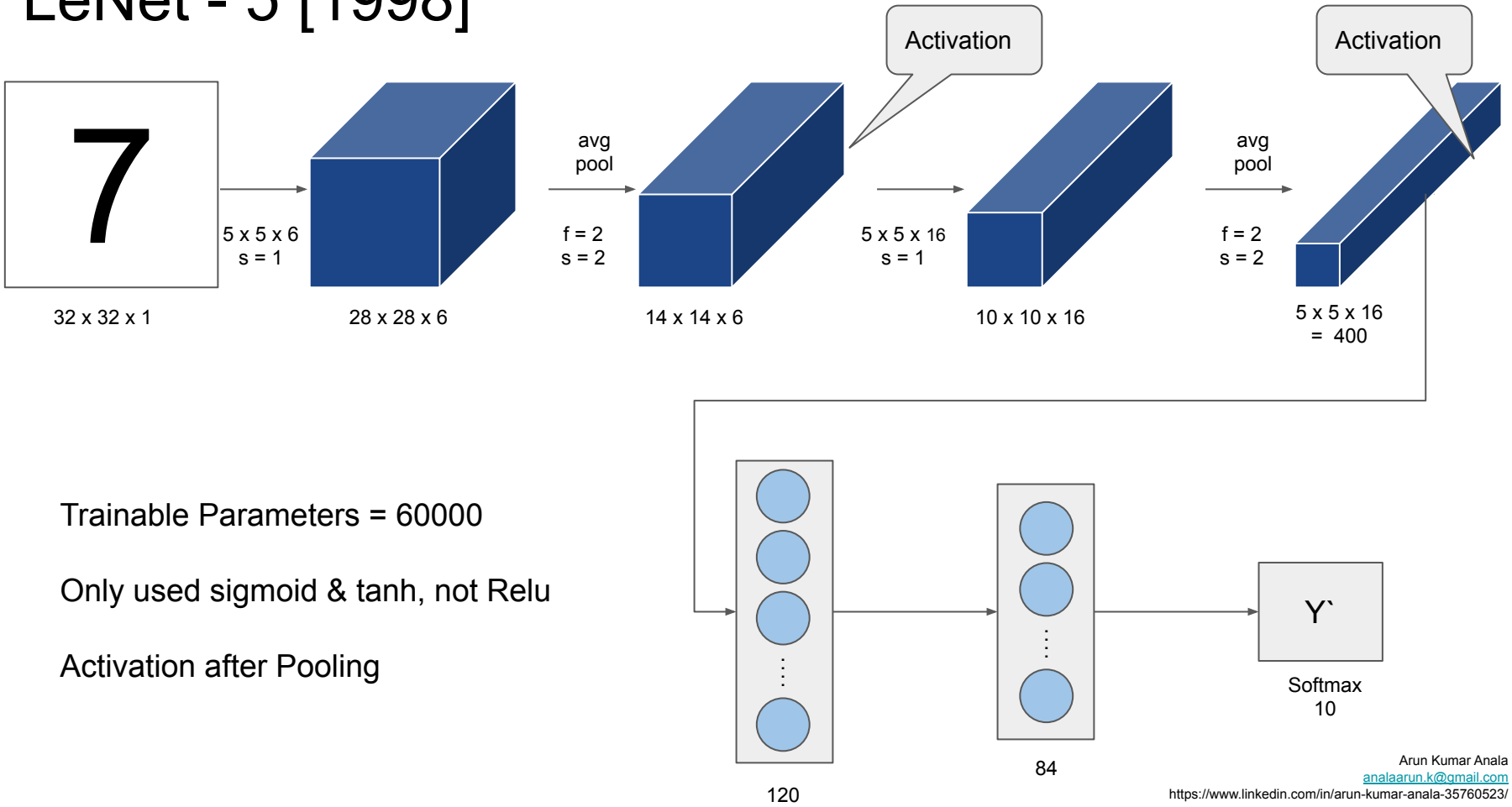
CNN

Week 2

Arun Kumar Anala
analaarun.k@gmail.com
https://www.linkedin.com/in/arun-kumar-anala-35760523/

LeNet - 5 [1998]



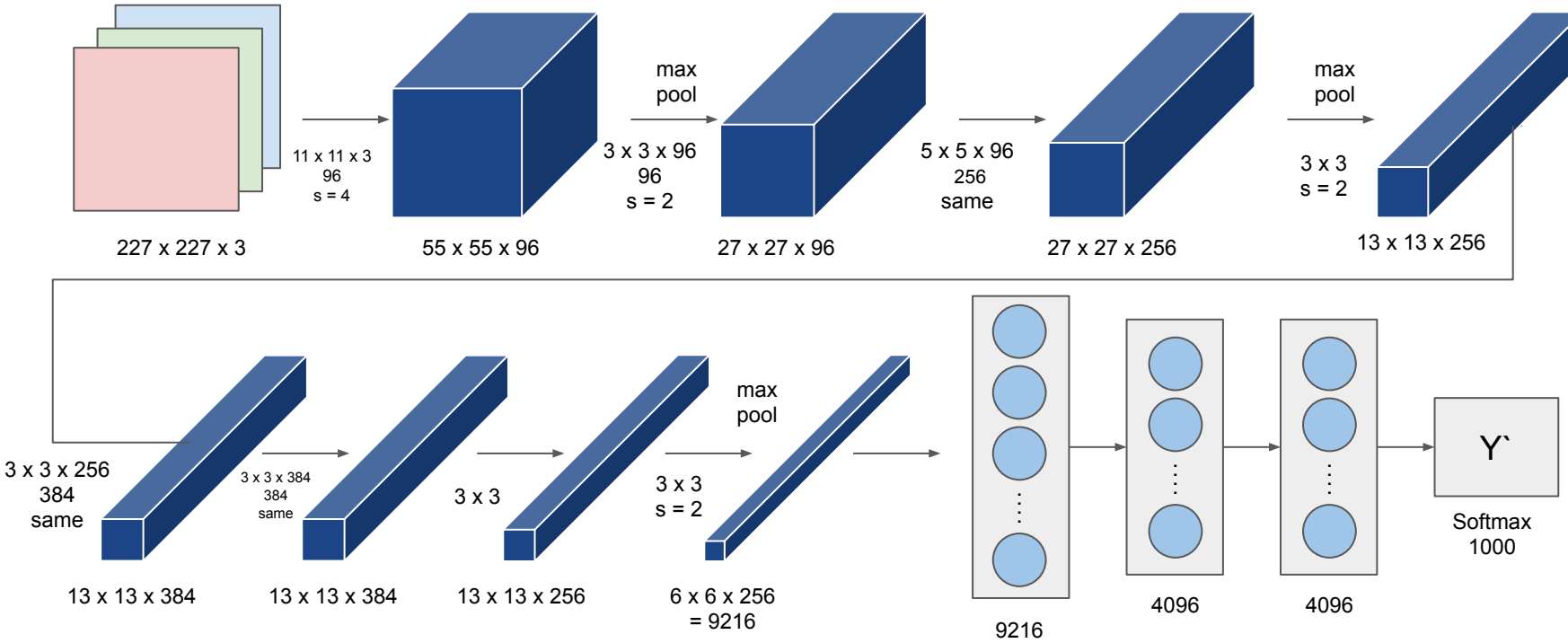
Trainable Parameters = 60000

Only used sigmoid & tanh, not Relu

Activation after Pooling

Arun Kumar Anala
analaarun.k@gmail.com
https://www.linkedin.com/in/arun-kumar-anala-35760523/

AlexNet [2012]



Trainable Parameters = 60 million

Used Relu

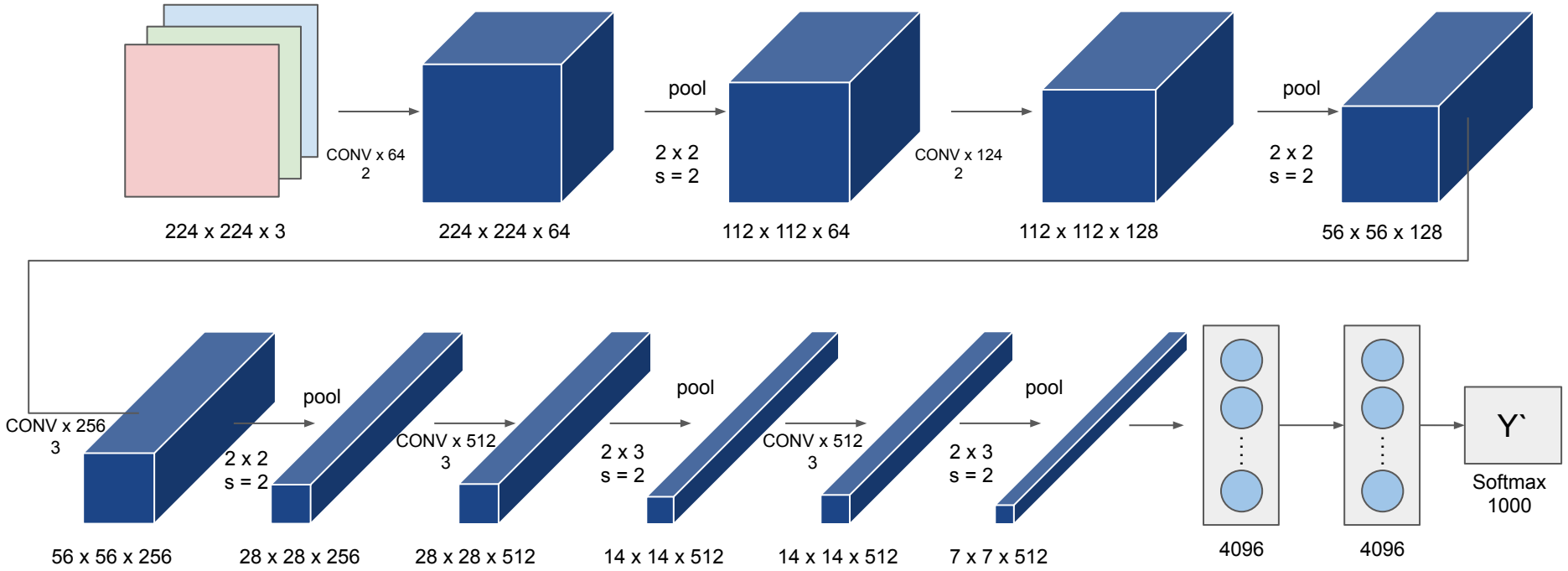
First Time Training on Multiple GPUs

Similar to LeNet but much bigger

Arun Kumar Anala
analaarun.k@gmail.com
https://www.linkedin.com/in/arun-kumar-anala-35760523/

VGG 16 [2015]

CONV = 3 x 3 , s = 1 , SAME
MAX POOL = 2 x 2, s = 2

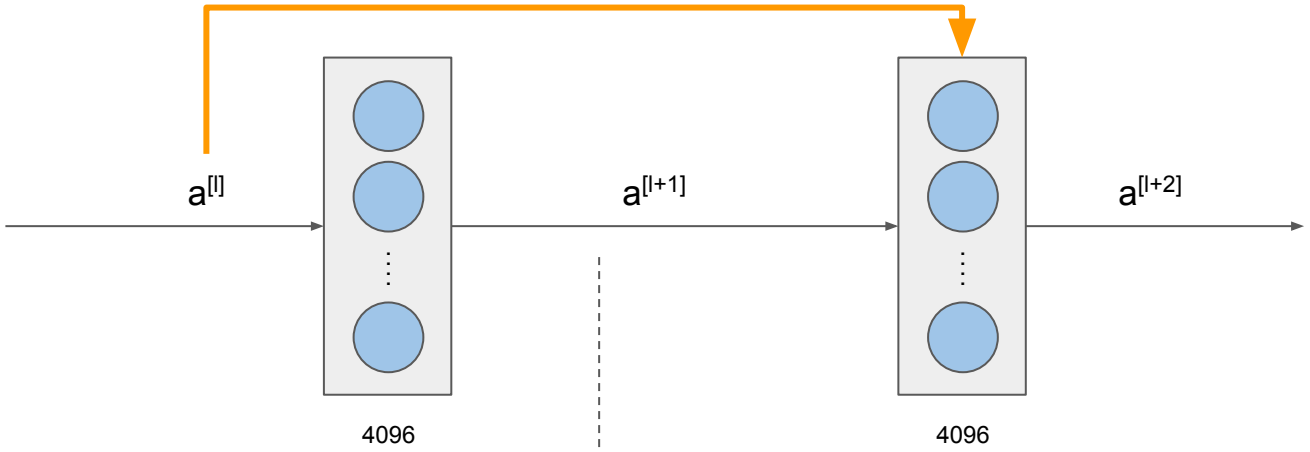


Trainable Parameters = 138 million
Pros - Uniformity
Goes Deeper by factor of 2
Multiple conv layers
3 x 3 filters across all layers

Cons - Bigger Network

Arun Kumar Anala
analaarun.k@gmail.com
https://www.linkedin.com/in/arun-kumar-anala-35760523/

ResNets [2015]



$z^{[l+1]} = w^{[l+1]} a^{[l]} + b^{[l+1]}$ $a^{[l+1]} = g(z^{[l+1]})$

$z^{[l+2]} = w^{[l+2]} a^{[l+1]} + b^{[l+2]}$ $a^{[l+2]} = g(z^{[l+2]})$

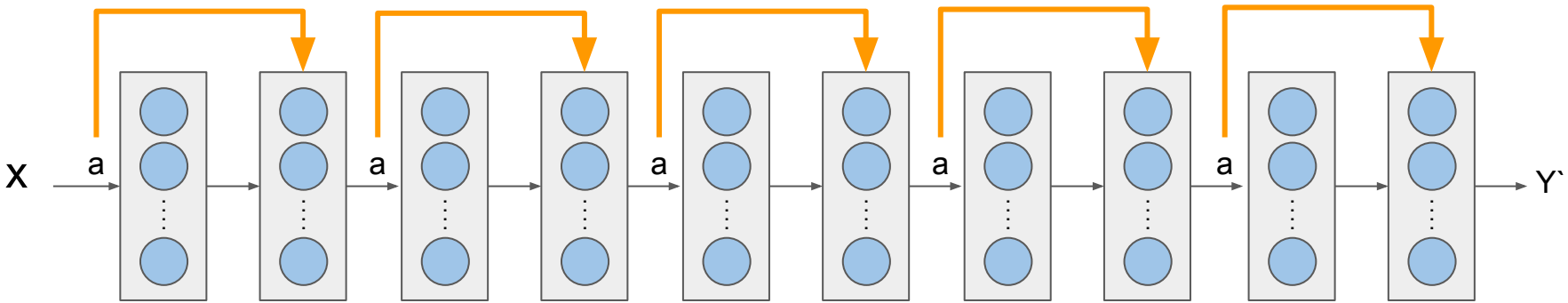
$$a^{[l+2]} = g(z^{[l+2]} + a^{[l]})$$

Also called Residual blocks / Skip Connection / Shortcuts

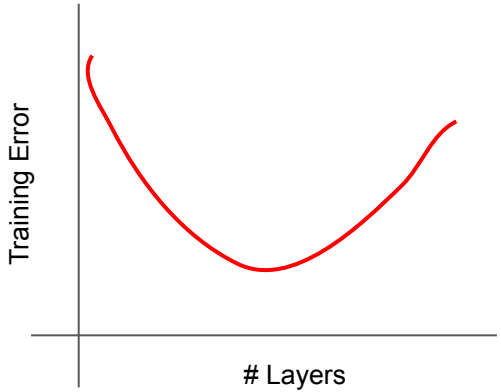
Using Residual blocks we can pass information much deeper into network, here train much deep networks,

Arun Kumar Anala
analaarun.k@gmail.com
https://www.linkedin.com/in/arun-kumar-anala-35760523/

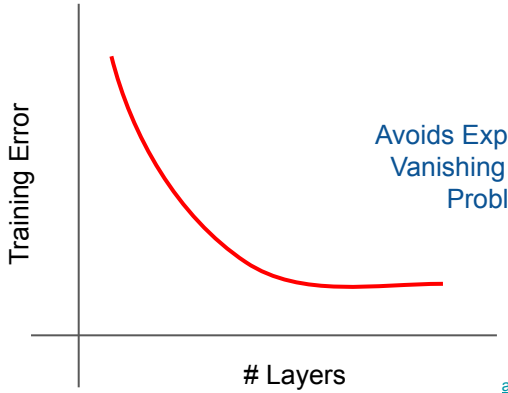
ResNets [2015]



Plain Network



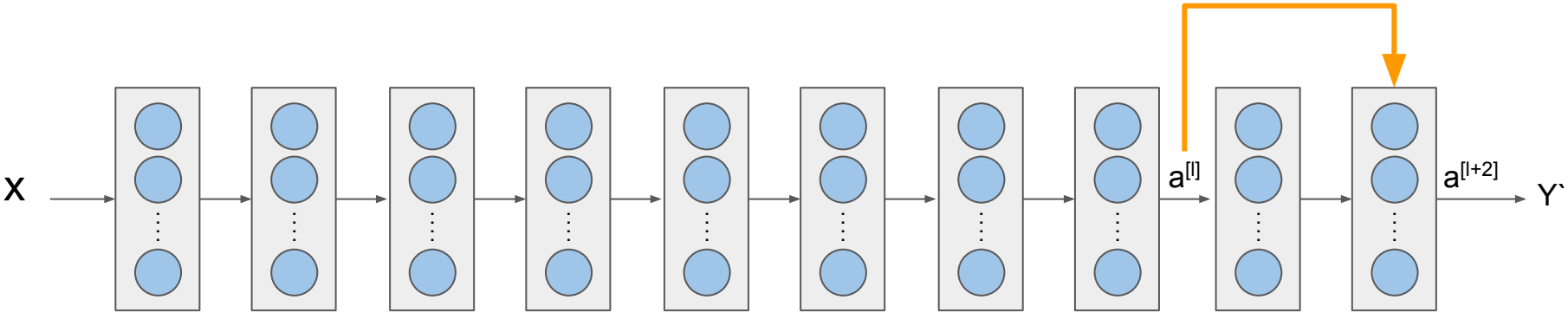
ResNet



Avoids Exploding and Vanishing Gradient Problems

Arun Kumar Anala
analaarun.k@gmail.com
https://www.linkedin.com/in/arun-kumar-anala-35760523/

ResNets [2015]



$z^{[l+2]} = w^{[l+2]} a^{[l+1]} + b^{[l+2]}$

$a^{[l+2]} = g(z^{[l+2]} + a^{[l]})$

$a^{[l+2]} = g(w^{[l+2]} a^{[l+1]} + b^{[l+2]} + a^{[l]})$ $w^{[l+2]} \sim 0,\ b^{[l+2]} \sim 0$

$a^{[l+2]} = g(a^{[l]}) = a^{[l]}$ Since Relu, $a^{[l]} > 0$

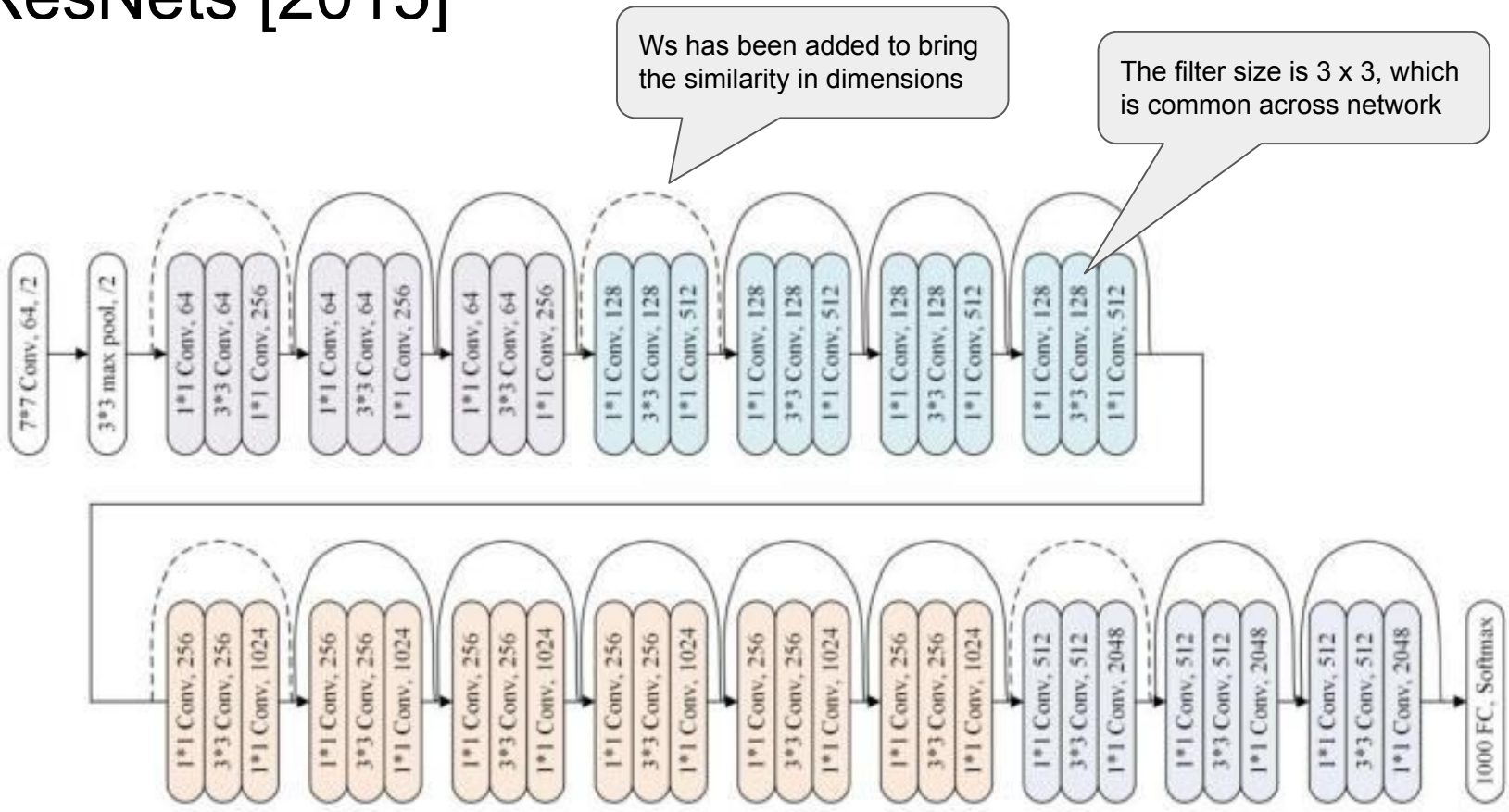
In case, dimension of $a^{[l+2]}$!= dimension of $a^{[l]}$
Ws can be used to match the dimension. Ws can be trainable parameters or fixed values like zero padding

$a^{[l+2]} = g(w^{[l+2]} a^{[l+1]} + b^{[l+2]} + W_s a^{[l]})$

Adding Residual Blocks does not hurts the performance of the plain network.

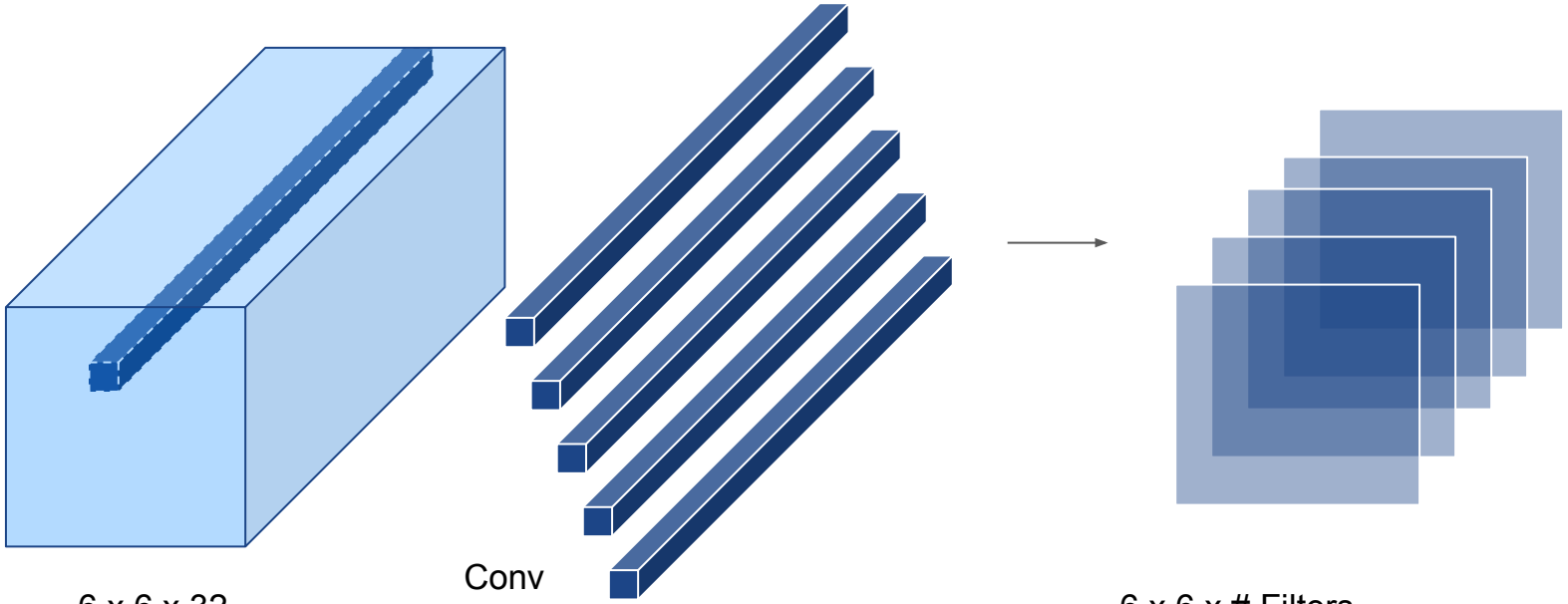
Arun Kumar Anala
analaarun.k@gmail.com
https://www.linkedin.com/in/arun-kumar-anala-35760523/

ResNets [2015]



Arun Kumar Anala
analaarun.k@gmail.com
https://www.linkedin.com/in/arun-kumar-anala-35760523/

1 x 1 Convolution

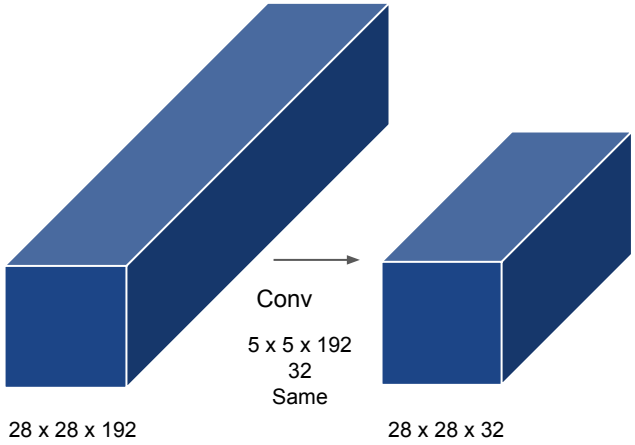


1 x 1 Convolution does not affects the height and width, but it's very useful in decreasing or increasing the depth.

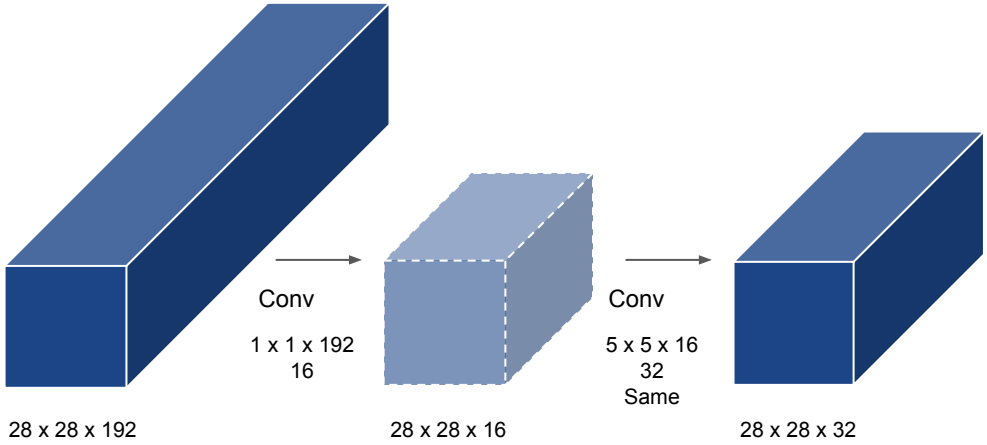
Or Just keep the output size same 6 x 6 x 32, so that we can bring in non-linearity.

Arun Kumar Anala
analaarun.k@gmail.com
https://www.linkedin.com/in/arun-kumar-anala-35760523/

1 x 1 Convolution



Total Multiplications = (28 x 28 x 32) x (5 x 5 x 192)
= 120 million

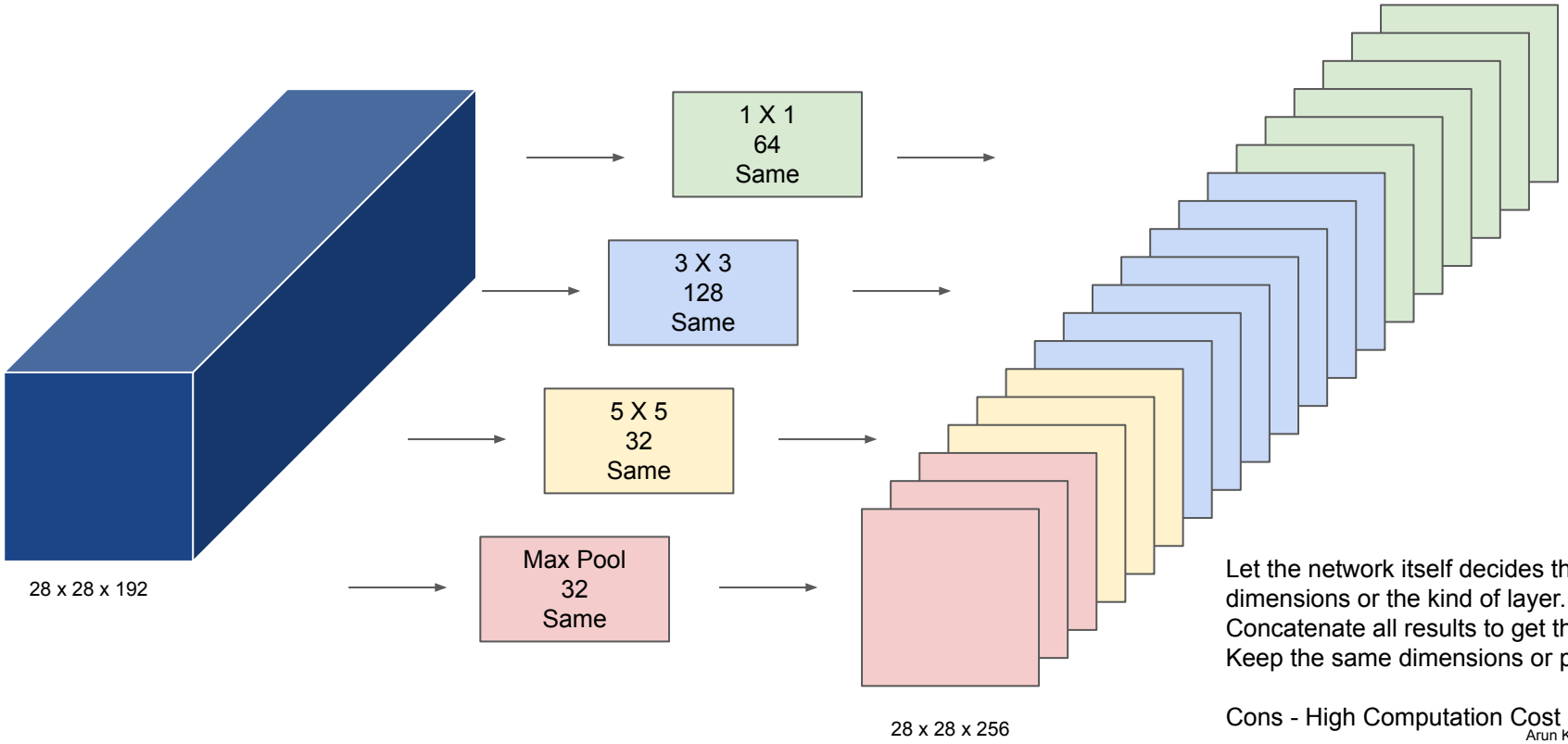


Total Multiplications = [(28 x 28 x 16) x (1 x 1 x 192)] + [(28 x 28 x 32) x (5 x 5 x 16)]
= 2.4 million + 10 million
= 12.4 million

Using 1 x 1 convolution, the cost of multiplication operations can be brought to 1/10th and here boost to performance

Arun Kumar Anala
analaarun.k@gmail.com
https://www.linkedin.com/in/arun-kumar-anala-35760523/

Inception [2014] , GoogleNet

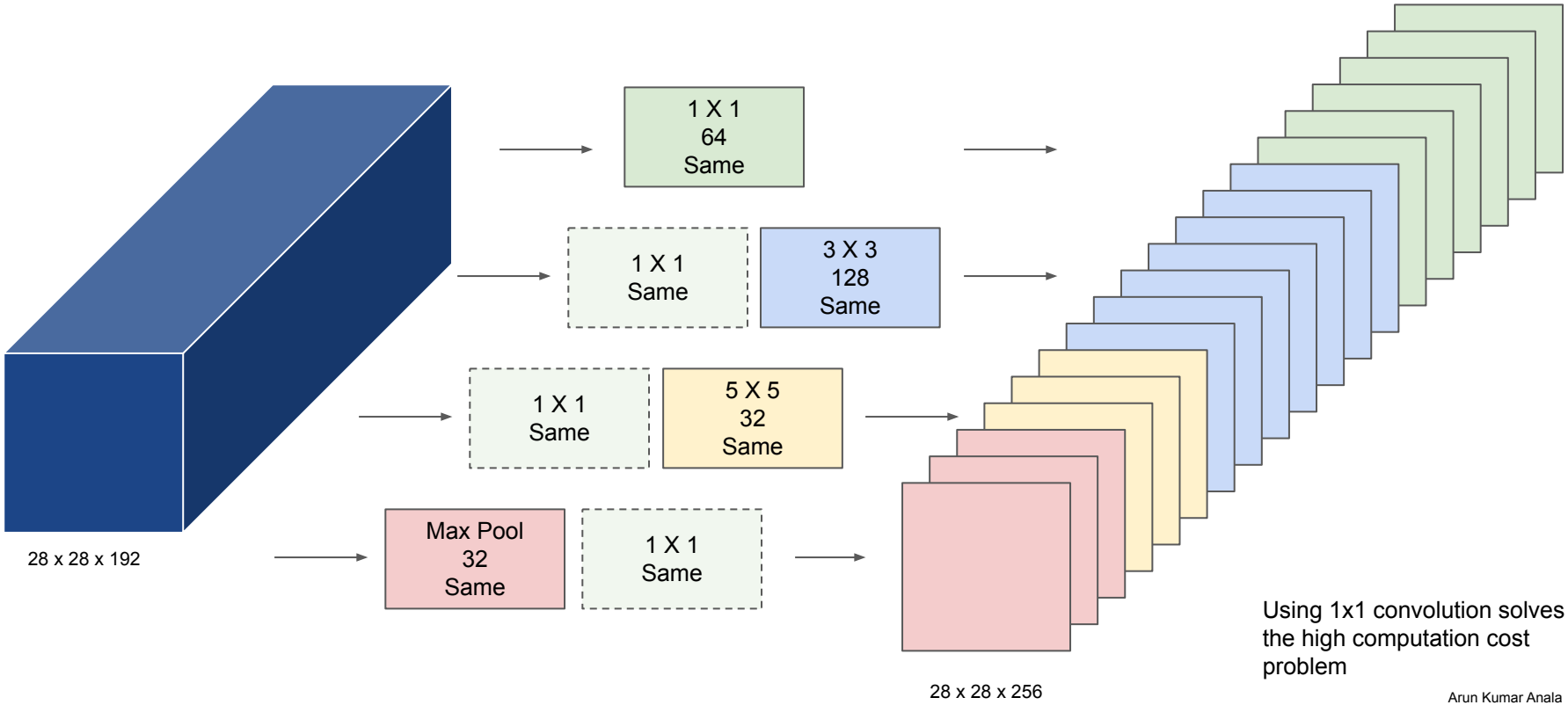


Let the network itself decides the filter dimensions or the kind of layer.
Concatenate all results to get the output.
Keep the same dimensions or padding

Cons - High Computation Cost

Arun Kumar Anala
analaarun.k@gmail.com
https://www.linkedin.com/in/arun-kumar-anala-35760523/

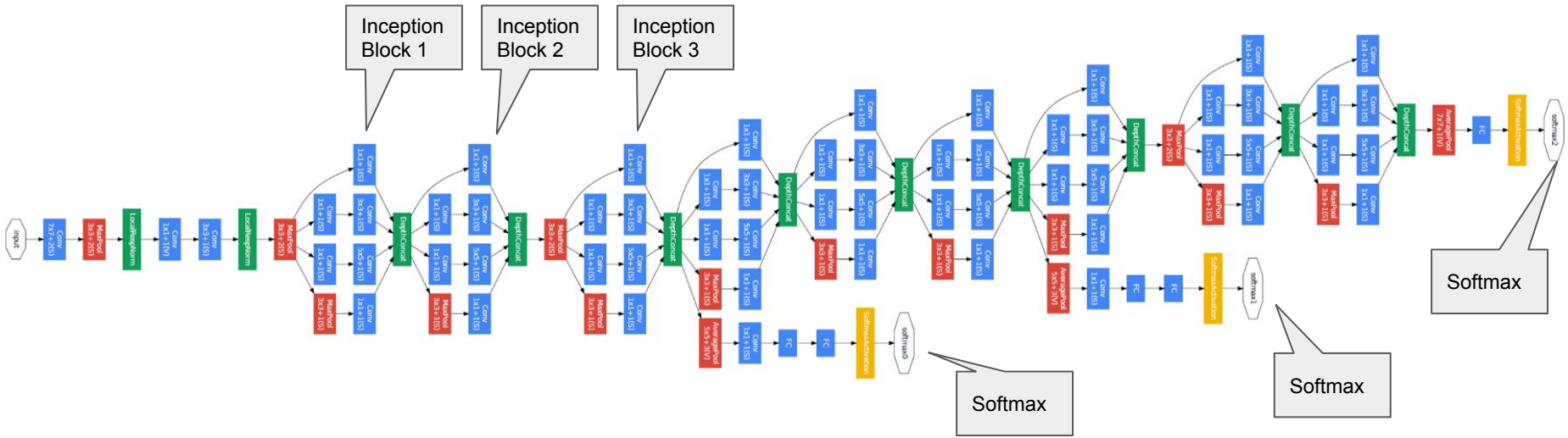
Inception [2014] , GoogleNet



Using 1x1 convolution solves the high computation cost problem

Arun Kumar Anala
analaarun.k@gmail.com
https://www.linkedin.com/in/arun-kumar-anala-35760523/

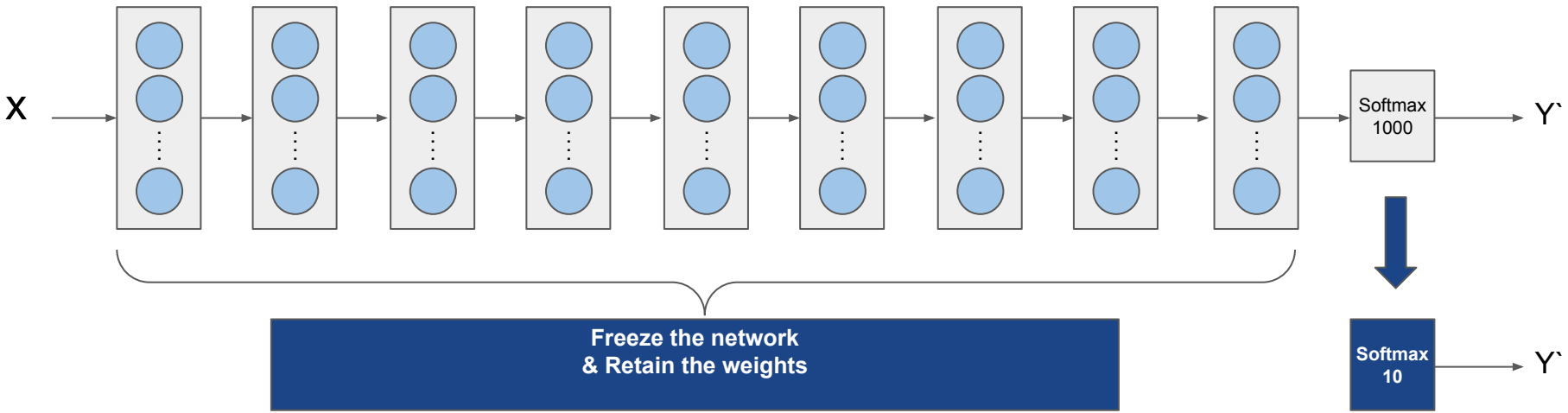
Inception [2014] , GoogleNet



Having Softmax, at multiple layers of network gives the opportunity to intermediate layers to learn better from the losses. This also have regularizing effect on intermediate layers and prevents overfitting.

Arun Kumar Anala
analaarun.k@gmail.com
https://www.linkedin.com/in/arun-kumar-anala-35760523/

Transfer Learning

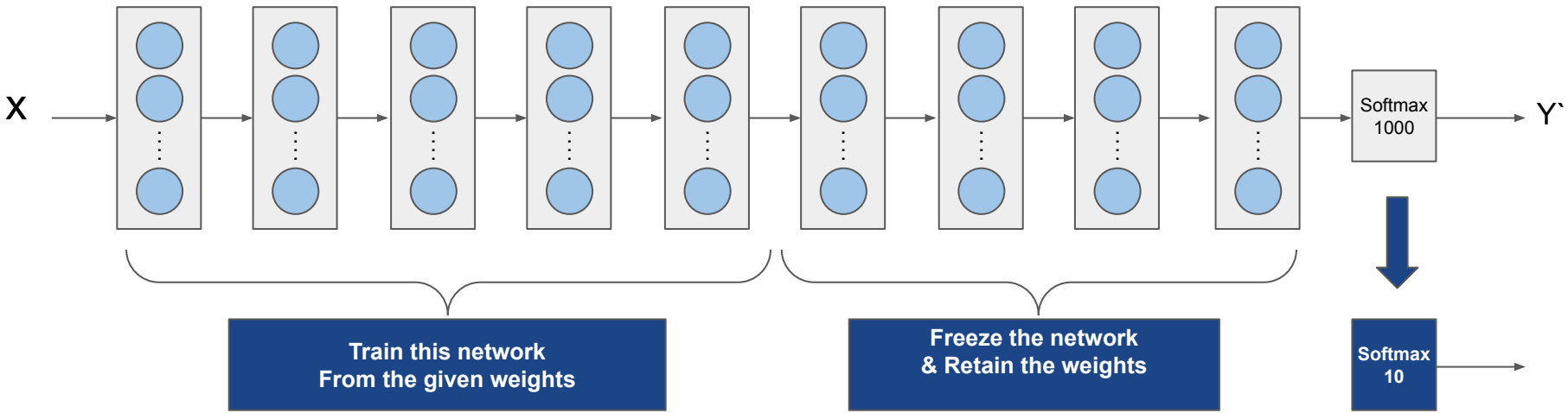


When data set is less

10 Classes to predict.

Arun Kumar Anala
analaarun.k@gmail.com
https://www.linkedin.com/in/arun-kumar-anala-35760523/

Transfer Learning



When data set is large

10 Classes to predict.

Arun Kumar Anala
analaarun.k@gmail.com
https://www.linkedin.com/in/arun-kumar-anala-35760523/

Data vs Hand Engineering

Little Data

Lots of Data

Object Detection

Image Recognition

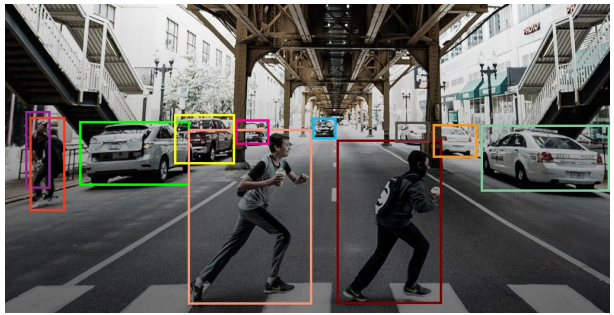
Speech Recognition

More Hand Engineering

Less Hand Engineering

Transfer Learning

Simpler Algorithms



Arun Kumar Anala
analaarun.k@gmail.com
https://www.linkedin.com/in/arun-kumar-anala-35760523/