# Test 02 – Networks

Arthur J. Redfern
arthur.redfern@utdallas.edu
Apr 03, 2019

---

# 0 Instructions

### Logistics

- Name: _____
- UT Dallas ID: _____

### Instructions

- There are 34 numbered questions with indicated point values that sum to 100
- Write all of your answers clearly on this test and turn it in
- No reference materials are allowed
- No help from others is allowed
- Correct answers in red

# 1 Test

### Design – Strategy [8 points]

Consider the following

A. A NN layer
B. A CNN layer
C. A RNN layer
D. An attention mechanism / layer

1. [4 points] Fill in A, B, C or D to make the following statements true (use each letter exactly once)

| | |
|---|---|
| B. CNN | maps input feature maps to output feature maps |
| C. RNN | maps an input feature vector and input state vector to an output feature vector and output state vector |
| A. NN | maps an input feature vector to an output feature vector |
| D. Attention | maps an input feature matrix and an input query vector / matrix to an output feature vector / matrix |

2. [3 points] Fill in B, C or D to make the following statements true (use each letter exactly once)

| | |
|---|---|
| C. RNN | exploits sequential structure in the data |
| D. Attention | exploits similarity with a query |
| B. CNN | exploits spatial structure in the data |

3. [1 points] Circle true or false for each of the following statements

| | |
|---|---|
| True / False | A linear (affine) layer is typically used to map an input feature vector to an output feature vector or class logit vector |

## Design – CNNs [21 points]

Consider the following image classification network (the filter tensors are specified as output channels x input channels x filter rows x filter cols / filter stride, the input feature map rows and cols are unspecified)

| | |
|---|---|
| Block A | |
| Residual connection | no |
| CNN style 2D conv | 64 x 3 x 7 x 7 / 2 |
| Max pooling | 3 x 3 / 2 |
| CNN style 2D conv | 128 x 64 x 1 x 1 |
| Block B | |
| Residual connection | yes |
| CNN style 2D conv | 32 x 128 x 1 x 1 |
| CNN style 2D conv | 32 x 32 x 3 x 3 |
| CNN style 2D conv | 128 x 32 x 1 x 1 |
| Block C | |
| Residual connection | no |
| Max pooling | 3 x 3 / 2 |
| CNN style 2D conv | 256 x 128 x 1 x 1 |
| Block D | |
| Residual connection | yes |
| CNN style 2D conv | 64 x 256 x 1 x 1 |

| | |
|---|---|
| CNN style 2D conv | 64 x 64 x 3 x 3 |
| CNN style 2D conv | 256 x 64 x 1 x 1 |
| Block E | |
| Residual connection | no |
| Global avg pool | rows x cols / rows x cols |
| Linear | classes x 256 |

4. [3 points] Which block(s) (A, B, C, D and / or E) are part of the …

| | |
|---|---|
| A | Tail |
| B, C and D | Body |
| E | Head |

5. [2 points] Which block(s) (A, B, C, D and / or E) are used for …

| | |
|---|---|
| A, B, C and D | Feature extraction |
| E | Prediction |

6. [1 points] Circle true or false for each of the following statements

True / False Is this network mathematically compatible with all non trivial input image row and col sizes?

7. [5 points] Assume blocks A, B, C and E are each included 1x, and block D is included 3x (i.e., the network looks like block A, B, C, D, D, D, E). What is the receptive field size at the output of the final block D?

(((1 + 3*(0 + 2 + 0) + 0)*2 – 1 + 2 + 0 + 2 + 0 + 0)*2 – 1 + 2)*2 – 1 + 6 = 75

8. [3 points] Compute the number of parameters and receptive field size at the output of the following layers / sequential layer combinations specified in terms of filters; CNN style 2D convolution is specified as output channels x input channels x filter rows x filter cols, depth wise convolution is specified as [output channels == input channels]: filter rows x filter cols; it's ok to leave the number of parameters in the form 1*2*3*4 instead of multiplying out to 24

| | | Receptive field size | Parameters |
|---|---|---|---|
| 8.1. | 32 x 32 x 5 x 5 | 1 + 4<br>= 5 | 32*32*5*5<br>= 25600 |
| 8.2. | 32 x 32 x 3 x 3<br>32 x 32 x 3 x 3 | 1 + 2 + 2<br>= 5 | 2*32*32*3*3<br>= 18432 |

| | | | |
|---|---|---|---|
| 8.3. | 32: 3 x 3 depth wise<br>32 x 32 x 1 x 1<br>32: 3 x 3 depth wise<br>32 x 32 x 1 x 1 | 1 + 0 + 2 + 0 + 2<br>= 5 | 2*32*3*3 +<br>2*32*32*1*1<br>= 2624 |

Consider an image classification network with the following structure: convolutional layers, global average pooling, linear layer. Assume that the network has been trained for M classes and the linear layer has a M x 256 weight matrix **F** and no bias term. Let's say I want to add a new class without retraining, so I input a batch of N examples of the new class to the network and get a batch of N features $\{\mathbf{f}_n\}_{n=0:N-1}$, each of size 256 x 1, at the output of global avg pooling. These N features are averaged together to get a single 256 x 1 class representative feature that we'll call $\mathbf{f} = (1/N)\ \Sigma_n\ \mathbf{f}_n$. The transpose of the class representative feature $\mathbf{f}^T$ is appended as an extra row to the linear layer weight matrix resulting in a (M + 1) x 256 weight matrix $[\mathbf{F}; \mathbf{f}^T]$ where ';' is used to denote that $\mathbf{f}^T$ is a row below matrix **F**.

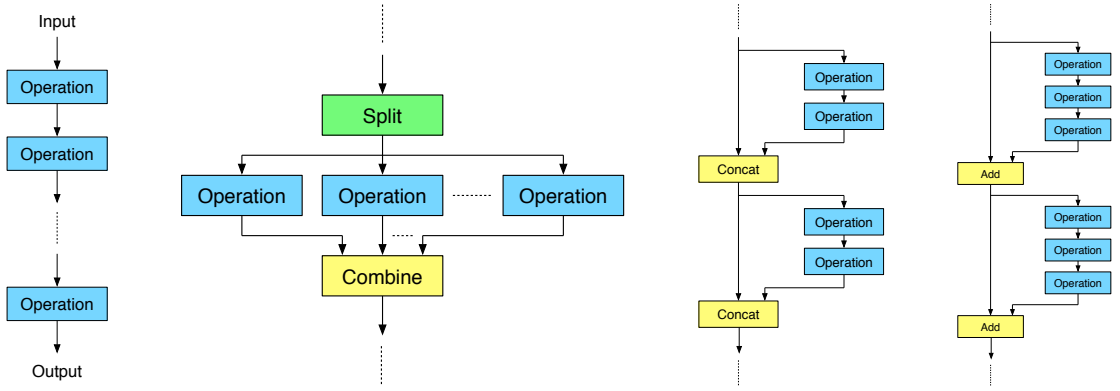
9. [3 points] Circle true or false for each of the following statements

True / False The inner product between **f** and the feature vector after global avg pooling for an input of the new class should ideally be large and positive

True / False The inner product between **f** and other columns of **F** should ideally be large and positive

True / False The magnitude of $\mathbf{f}^T$ should ideally be similar to other rows of **F** assuming that classes are equally likely

Consider the following network building blocks



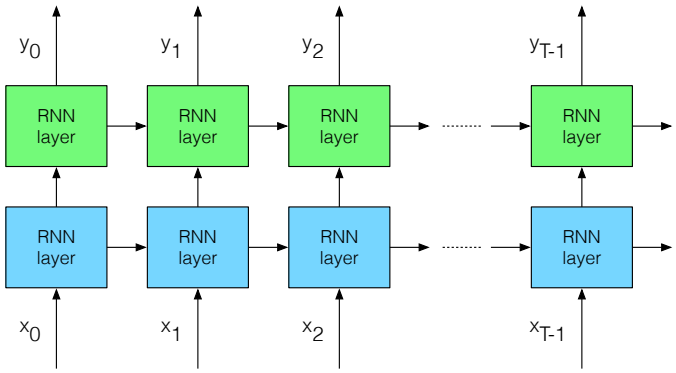

A B C D

10. [4 points] Which building block (A, B, C or D) is of the following type (use each letter exactly once)

D. Residual
B. Parallel
A. Sequential / serial
C. Densely connected

## Design – RNNs [4 points]

Consider the following unrolled RNN structure




11. [4 points] Circle true or false for each of the following statements

True / False The same input and input state vector → output and output state vector transformation is used for all of the green boxes

True / False The same input and input state vector → output and output state vector transformation is used for all of the green and blue boxes

True / False This is a bi directional RNN

True / False This is a pyramidal RNN

## Design – Attention [12 points]

Consider an encoder RNN, attention mechanism and decoder RNN structure. N encoded features of size K x 1 are placed shoulder to shoulder in the K x N matrix $\mathbf{X}$ and $\mathbf{q}_m$ is a K x 1 decoder query at position m. A dot product and soft max are used to determine a N x 1 attention distribution as $\mathbf{p}_m = \text{softmax}(\mathbf{X}^T \mathbf{q}_m)$.

12. [4 points] Write an equation for the K x 1 attention output $\mathbf{y}_m$ using the above definitions

$\mathbf{y}_m = \mathbf{X}\,\text{softmax}(\mathbf{X}^T \mathbf{q}_m)$

$= X\ p_m$

13. [2 points] Circle true or false for each of the following statements

True / False The attention distribution is dependent on the individual magnitudes of the N encoded features

True / False The attention distribution is dependent on the individual angles between the N encoded features and the query

Consider a simplified single headed self attention operation that maps N input features of size K x 1 placed shoulder to shoulder in a K x N matrix **X** to N output features of size K x 1 placed shoulder to shoulder in a K x N matrix **Y** via the following set of equations where **W**'s are K x K learned weight matrices and the soft max is applied individually to columns of the matrix

$$Q = W_Q X$$
$$K = W_K X$$
$$V = W_V X$$
$$Y = V\ \text{softmax}(K^T Q)$$

14. [4 points] After all of the weight matrices **W** have been trained, what term can I compute once to reduce the overall computation of the self attention operation during subsequent uses in testing / deployment?

$$Y = V\ \text{softmax}(K^T Q)$$
$$= V\ \text{softmax}(X^T W_K^T W_Q X)$$
$$= V\ \text{softmax}(X^T W_{KQ} X)$$

where $W_{KQ} = W_K^T W_Q$

Computing $W_{KQ}$ saves 1 matrix matrix multiplication per subsequent use

15. [2 points] What does this imply with respect to the number of unique parameters that control the input / output mapping?

This implies that the number of unique parameters that controls the input / output mapping is actually $2K^2$ and not $3K^2$

## Training [25 points]

16. [1 points] Circle true or false for each of the following statements

True / False Generalization is not a problem if the mean and variance of the training and testing data are the same

17. [3 points] Order the following image labeling tasks 1, 2 and 3 from least complex (1) to most complex (3) for a human:

2 Multiple object detection
1 Classification
3 Depth estimation

18. [2 points] List the arguably 2 most important / commonly used network components / modifications that have allowed for the training of arbitrarily deep networks

Batch normalization, residual connections

19. [3 points] Consider a CNN style 2D convolution layer composed of the following components: convolution, bias and ReLU nonlinearity. List the components that are present during training if batch normalization is included with the layer

Convolution, batch normalization and ReLU (bias is handled by batch normalization)

20. [3 points] What type of moving average does batch normalization use to track time varying weights during training?

Exponential moving average

21. [3 points] Circle true or false for each of the following statements

True / False Residual connections can be used with CNNs
True / False Residual connections can be used with RNNs
True / False Residual connections can be used with self attention based networks

22. [3 points] Consider a network that ends with a large fully connected layer with output size N followed by a linear layer with output size C equal to the number of classes. If dropout with 50% is used to regularize the large fully connected layer, how many final features at a time does the linear layer learn to use to predict classes per batch?

N / 2

Consider a classification network with $l_2$ weight decay contributing a term to the error:

$$\begin{aligned} \text{error} &= \text{cross entropy error} + l_2 \text{ weight error} \\ &= \text{cross entropy error} + \lambda \Sigma_k h^2(k) \end{aligned}$$

where $\{h(k)\}_{k=0:K-1}$ is the set of all weights and $\lambda$ is a scalar.

23. [2 points] Circle true or false for each of the following statements

True / False $l_2$ weight decay penalizes large weights more than small weights

True / False $l_2$ weight decay acts as a regularizer

The Adam method for stochastic weight optimization computes **s**, an exponential moving average of the gradient, and **r**, an exponential moving average of the elementwise square of the gradient, both across batches. After correcting for a bias in the calculation, updates to weights are

$$\mathbf{h} \leftarrow \mathbf{h} - \alpha\, \mathbf{s}./(\mathbf{r}^{.1/2} + \delta)$$

with ./ used to indicate elementwise division and .1/2 used to indicate elementwise square root. $\delta$ is a small value.

24. [2 points] Circle true or false for each of the following statements

True / False Elementwise division by $\mathbf{r}^{.1/2}$ reduces movements more in small gradient directions than in large gradient directions

True / False The exponential moving average creates a momentum effect across batches

25. [3 points] In data synchronous parallel training consider a system with 1 master, 32 worker machines and a batch size of 32 per worker machine. What is the effective batch size per weight update made by the master machine?

32*32 = 1024

## Implementation [30 points]

26. [4 points] Let $x_{16}$ and $y_{16}$ each be 16 bit unsigned integers. Show that $x_{16}*y_{16}$ can be implemented with four 8 bit multiplications, three shifts and three adds. Define 8 bit terms as necessary, your result should be a single equation in the form

$x_{16}*y_{16}$ = <your equation here with four 8 bit multiplications, three shifts and three adds>

$x_{16} = 2^8\, x_8^{hi} + x_8^{lo}$

$y_{16} = 2^8\, y_8^{hi} + y_8^{lo}$

$x_{16}\, y_{16} = (2^8\, x_8^{hi} + x_8^{lo})\, (2^8\, y_8^{hi} + y_8^{lo})$

$= 2^{16}\, x_8^{hi}\, y_8^{hi} + 2^8\, x_8^{hi}\, y_8^{lo} + 2^8\, x_8^{lo}\, y_8^{hi} + x_8^{lo}\, y_8^{lo}$

Say I quantize 32 bit IEEE 754 float (1 bit sign, 23 bits significand, 8 bits exponent) to 16 bit bfloat (1 bit sign, 7 bits significand, 8 bits exponent).

27. [3 points] What is the approximate reduction in bits of precision?
    23 bits → 7 bits (16 bit reduction in precision)

28. [3 points] What is the approximate reduction in bits of range?
    8 bits → 8 bits (no / 0 bit reduction in range)

29. [3 points] Let input matrices **A** and **B** and output matrix **C** all be stored off device and the goal is to compute **C** = **A***B on device. If **A** fits fully on device, how many times do **B** and **C** each need to be moved between off device and on device?
    1x each (**B** from off device to on device, **C** from on device to off device)

Consider inputs

L = transistor feature size
V = voltage

and approximate semiconductor device physics values

C = capacitance per transistor ($\propto$ L)
D = area density ($\propto 1/L^2$)
E = energy per transistor use ($\propto CV^2$)
f = frequency ($\propto 1/L$)
P = power per area ($\propto$ DEf)

where $\propto$ means "proportional to".

30. [4 points] How does the power per area P change if the transistor feature size L shrinks by 1/2 (i.e., 2 generations of process scaling) but the voltage V stays the same?
    C → C/2
    D → 4D
    E → E/2
    f → 2f
    P → 4P (so power per area increases by 4x)

31. [1 points] Circle true or false for each of the following statements

True / False It's a good idea to put high level algorithms that don't belong to a standard in gates (fixed circuit)

Consider CNN style 2D convolution with

| | |
|---|---|
| Input feature maps ($N_i$ x $L_r$ x $L_c$) of size | 16 x 256 x 512 |
| Row pad = col pad of size | 2 |
| Filter ($N_o$ x $N_i$ x $F_r$ x $F_c$) | 32 x 16 x 3 x 3 |
| Output feature maps ($N_o$ x $M_r$ x $M_c$) of size | 32 x 256 x 512 |

that's lowered to a matrix matrix multiplication problem and computed via repeated use of a matrix matrix multiplication primitive that can only multiply 32 x 32 matrix tiles (i.e., problem tiles smaller than 32 x 32 are zero padded to 32 x 32 by the hardware). Define efficiency as

efficiency = MACs required by the problem / MACs computed by the hardware

32. [5 points] What is the efficiency of the above described hardware applied to the above described problem?

Using BLAS M, N and K notation for the problem and hardware

| | | |
|---|---|---|
| $M_{problem}$ | = $N_o$ | = 32 |
| $N_{problem}$ | = $M_r$*$M_c$ | = 256*512 |
| $K_{problem}$ | = $N_i$*$F_r$*$F_c$ | = 16*3*3 |
| $MAC_{problem}$ | = $M_{problem}$*$N_{problem}$*$K_{problem}$ | = 32*256*512*16*3*3 |
| | | |
| $M_{hw}$ | = ceil($M_{problem}$, 32) | = $M_{problem}$ |
| $N_{hw}$ | = ceil($N_{problem}$, 32) | = $N_{problem}$ |
| $K_{hw}$ | = ceil($K_{problem}$, 32) | = 16*3*3 + 16 = 16*10 |
| $MAC_{hw}$ | = $M_{hw}$*$N_{hw}$*$K_{wn}$ | = 32*256*512*16*10 |
| | | |
| Efficiency | = $MAC_{problem}$ / $MAC_{hw}$ | = (3*3)/10 = 0.90 or 90 % |

33. [3 points] To reduce the cost of a calculation, which of the following are generally used strategies? Circle true or false for each of the following statements

True / False Exploiting that different operations have different costs and doing less of the higher cost operation at the expense of doing more of the lower cost operation

True / False Creating intermediate terms that can be re used

True / False Recursively applying the intermediate term re use strategy

34. [4 points] Say an operation requires data movement that takes time $T_{datamove}$ and compute that takes time $T_{compute}$. If the startup time and finish times are negligible and ping pong buffers can be used to maximally overlap data movement and compute, what is the total time for this operation?

$T_{operation} = \max(T_{datamove}, T_{compute})$