Capsule Networks

Dynamic Routing Between Capsules

Louis (Yiqing) Luo

July 6th, 2018

Outline

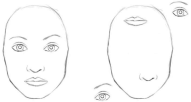
1. Capsule Network Intuition
   - Capsule Network - Intuition
   - Better Representation of Images: Inverse Graphics
2. Capsules
   - What are capsules
   - Capsule vs Neurons
   - Properties of capsules
3. Capsule Network
   - Capsule Network Encoder Architecture
   - Capsule Network Decoder Architecture
4. Experiments and Results
   - MNIST
   - multiMNIST
   - CIFAR10

Outline

1. Capsule Network Intuition
   - Capsule Network - Intuition
   - Better Representation of Images: Inverse Graphics
2. Capsules
   - What are capsules
   - Capsule vs Neurons
   - Properties of capsules
3. Capsule Network
   - Capsule Network Encoder Architecture
   - Capsule Network Decoder Architecture
4. Experiments and Results
   - MNIST
   - multiMNIST
   - CIFAR10

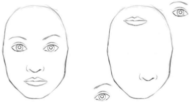
Capsule Network - Intuition

- Drawbacks of CNN currently constructed
  1. Sub-sampling loses the precise spatial relationships between higher-level parts such as a nose and a mouth. The precise spatial relationships are needed for identity recognition
     - But overlapping the sub-sampling pools mitigates this.
  2. They cannot extrapolate their understanding of geometric relationships to radically new viewpoints.
  3. Discard (invariant) instead of disentangle (equivariant). Knowledge should be invariant, not activities inside networks.



Capsule Network - Intuition

- Drawbacks of CNN currently constructed
  1. Sub-sampling loses the precise spatial relationships between higher-level parts such as a nose and a mouth. The precise spatial relationships are needed for identity recognition
     - But overlapping the sub-sampling pools mitigates this.
  2. They cannot extrapolate their understanding of geometric relationships to radically new viewpoints.
  3. Discard (invariant) instead of disentangle (equivariant). Knowledge should be invariant, not activities inside networks.



Hinton: The pooling operation used in convolutional neural networks is a big mistake and the fact that it works so well is a disaster.

Outline

1. Capsule Network Intuition
   - Capsule Network - Intuition
   - Better Representation of Images: Inverse Graphics

2. Capsules
   - What are capsules
   - Capsule vs Neurons
   - Properties of capsules

3. Capsule Network
   - Capsule Network Encoder Architecture
   - Capsule Network Decoder Architecture

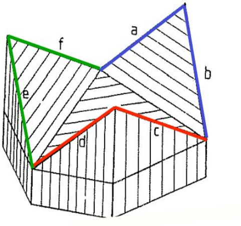
4. Experiments and Results
   - MNIST
   - multiMNIST
   - CIFAR10

Better Representation of Images: Inverse Graphics

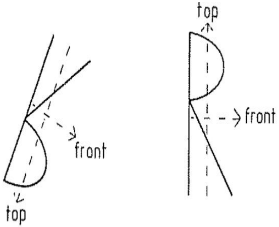
- Hinton claims that brain deconstruct hierarchy representation of visual information and tries to match patterns and relationships already learned.
  - representation of objects in brain does not depend on view angles.
- 3D graphics: relationships between 3D objects can be represented by poses
- Hinton: correct classification and recognition requires preservation of hierarchy pose relationship between objects

Better Representation of Images: Inverse Graphics

- Inverse Graphics is widely employed in computer vision.
- Uses hierarchical models in which spatial structure is modeled by matrices that represent the transformation from a coordinate frame embedded in the whole to a coordinate frame embedded in each part
- matrices are viewpoint invariant.



(a)



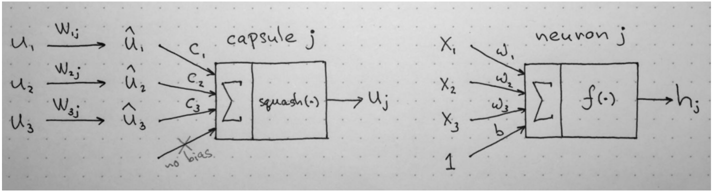
(b)

Outline

1. Capsule Network Intuition
   - Capsule Network - Intuition
   - Better Representation of Images: Inverse Graphics
2. Capsules
   - What are capsules
   - Capsule vs Neurons
   - Properties of capsules
3. Capsule Network
   - Capsule Network Encoder Architecture
   - Capsule Network Decoder Architecture
4. Experiments and Results
   - MNIST
   - multiMNIST
   - CIFAR10

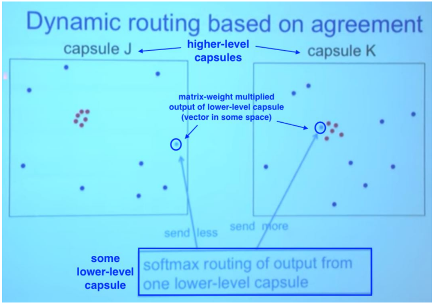
What are capsules

- each capsule will learn to recognize an implicitly defined visual entity over a limited domain of viewing conditions and deformations
- Outputs:
  1. probability that the entity is within its limited domain (length of the vector does note change)
  2. a set of instantiation parameters (eg. pose, lighting, possible deformations relative to implicitly defined canonical version of entities) (orientation of the vector changes)



Coincidence Filtering

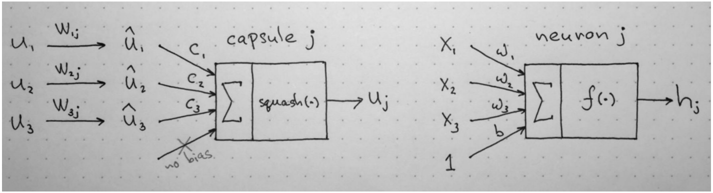
- capsules receive multi-dimensional prediction vectors from capsules in layer below and look for tight predictions
- Outputs can also be described as:
  1. probability that the entity exist within its domain
  2. Center of gravity of the cluster = generalized pose
- good at filtering noise because coincidences rarely happens in high dimension space

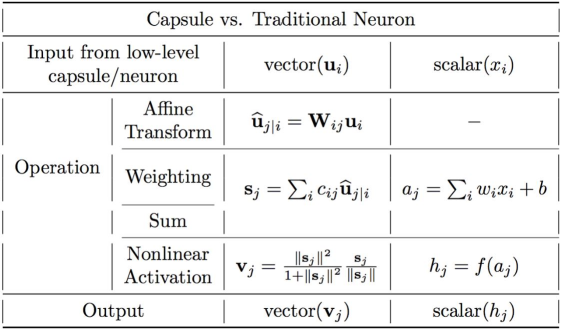

Dynamic routing based on agreement
capsule J
higher-level capsules
capsule K
matrix-weight multiplied output of lower-level capsule (vector in some space)
send less
send more
some lower-level capsule
softmax routing of output from one lower-level capsule

Outline

1. Capsule Network Intuition
   - Capsule Network - Intuition
   - Better Representation of Images: Inverse Graphics
2. Capsules
   - What are capsules
   - Capsule vs Neurons
   - Properties of capsules
3. Capsule Network
   - Capsule Network Encoder Architecture
   - Capsule Network Decoder Architecture
4. Experiments and Results
   - MNIST
   - multiMNIST
   - CIFAR10

Capsule vs Neurons





Capsule vs. Traditional Neuron			
Input from low-level capsule/neuron		vector($\mathbf{u}_i$)	scalar(x_i)
Operation	Affine Transform	$\widehat{\mathbf{u}}_{j\|i} = \mathbf{W}_{ij}\mathbf{u}_i$	–
	Weighting	$\mathbf{s}_j = \sum_i c_{ij}\widehat{\mathbf{u}}_{j\|i}$	$a_j = \sum_i w_i x_i + b$
	Sum		
	Nonlinear Activation	$\mathbf{v}_j = \frac{\|\mathbf{s}_j\|^2}{1+\|\mathbf{s}_j\|^2}\frac{\mathbf{s}_j}{\|\mathbf{s}_j\|}$	$h_j = f(a_j)$
Output		vector($\mathbf{v}_j$)	scalar(h_j)

Matrix Multiplication on Input Vectors

- inputs u_1, u_2, u_3 contains probability of features and their poses
- eg. for detecting a nose from a face:
    - W_{ij} encodes relationship from i_{th} feature (nose) to j_{th} feature (face)

$$\widehat{\mathbf{u}}_{j|i} = \mathbf{W}_{ij}\mathbf{u}_i$$

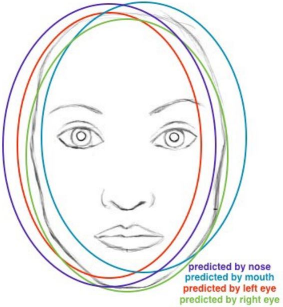
Scalar Weighting

- measures the affinity between low level capsules and high level capsules
- c_{ij} is adjusted each iteration by dynamic routing

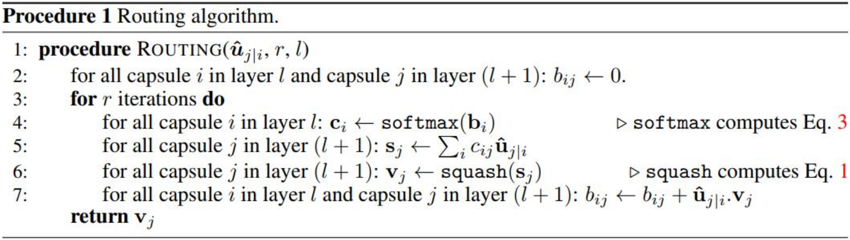
$$\mathbf{s}_j = \sum_i c_{ij} \widehat{\mathbf{u}}_{j|i}$$

Scalar Weighting - Dynamic Routing

- decides how to send output vector to higher level capsule j by changing c_{ij}
- important properties of c_{ij}
  1. c_{ij} non-negative scalar
  2. $\Sigma_j c_{ij} = 1$ for all i
  3. # of weights for the vector c_i = # of higher level capsules
  4. determined by iterative dynamic routing algorithm

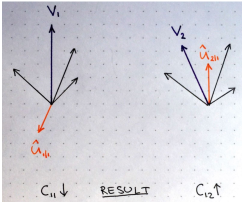


Dynamic Routing - Algorithm



Procedure 1 Routing algorithm.

```
procedure ROUTING($\hat{\mathbf{u}}_{j|i}$, $r$, $l$)
    for all capsule $i$ in layer $l$ and capsule $j$ in layer $(l+1)$: $b_{ij} \leftarrow 0$.
    for $r$ iterations do
        for all capsule $i$ in layer $l$: $\mathbf{c}_i \leftarrow \mathtt{softmax}(\mathbf{b}_i)$    ▷ softmax computes Eq. 3
        for all capsule $j$ in layer $(l+1)$: $\mathbf{s}_j \leftarrow \sum_i c_{ij}\hat{\mathbf{u}}_{j|i}$
        for all capsule $j$ in layer $(l+1)$: $\mathbf{v}_j \leftarrow \mathtt{squash}(\mathbf{s}_j)$    ▷ squash computes Eq. 1
        for all capsule $i$ in layer $l$ and capsule $j$ in layer $(l+1)$: $b_{ij} \leftarrow b_{ij} + \hat{\mathbf{u}}_{j|i}.\mathbf{v}_j$
    return $\mathbf{v}_j$
```



Squash for Non-Linearity

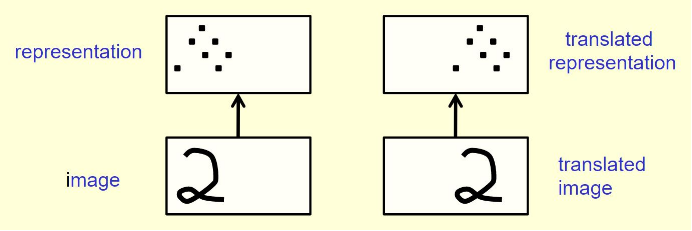
$$\mathbf{v}_j = \underbrace{\boxed{\frac{\|\mathbf{s}_j\|^2}{1+\|\mathbf{s}_j\|^2}}}_{\text{additional “squashing”}} \underbrace{\boxed{\frac{\mathbf{s}_j}{\|\mathbf{s}_j\|}}}_{\text{unit scaling}}$$

Outline

1. Capsule Network Intuition
   - Capsule Network - Intuition
   - Better Representation of Images: Inverse Graphics
2. Capsules
   - What are capsules
   - Capsule vs Neurons
   - Properties of capsules
3. Capsule Network
   - Capsule Network Encoder Architecture
   - Capsule Network Decoder Architecture
4. Experiments and Results
   - MNIST
   - multiMNIST
   - CIFAR10

Properties of capsules

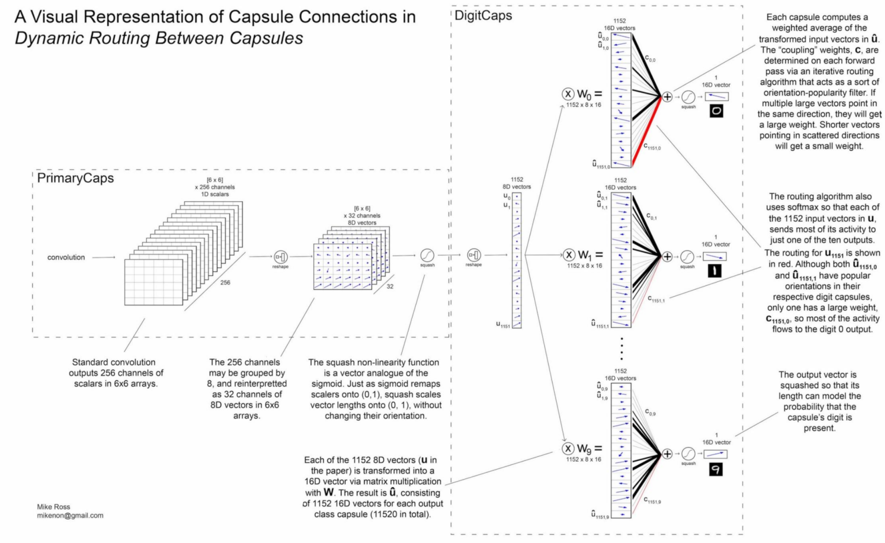
1. probability = invariant
   - does note change for possible appearances in manifold within limited domain covered by capsule
2. instantiation parameters = equivariant
   - change in viewing conditions, position etc. change correspondingly over the appearance manifold
     - face is centered around the nose
     - nose is 10 times smaller than the face
     - same orientation in space



Outline

1. Capsule Network Intuition
   - Capsule Network - Intuition
   - Better Representation of Images: Inverse Graphics
2. Capsules
   - What are capsules
   - Capsule vs Neurons
   - Properties of capsules
3. **Capsule Network**
   - **Capsule Network Encoder Architecture**
   - Capsule Network Decoder Architecture
4. Experiments and Results
   - MNIST
   - multiMNIST
   - CIFAR10

Capsule Network Encoder Architecture



A Visual Representation of Capsule Connections in *Dynamic Routing Between Capsules*

Standard convolution outputs 256 channels of scalars in 6x6 arrays.

The 256 channels may be grouped by 8, and reinterpretted as 32 channels of 8D vectors in 6x6 arrays.

The squash non-linearity function is a vector analogue of the sigmoid. Just as sigmoid remaps scalers onto (0,1), squash scales vector lengths onto (0, 1), without changing their orientation.

Each of the 1152 8D vectors ($\mathbf{u}$ in the paper) is transformed into a 16D vector via matrix multiplication with $\mathbf{W}$. The result is $\hat{\mathbf{u}}$, consisting of 1152 16D vectors for each output class capsule (11520 in total).

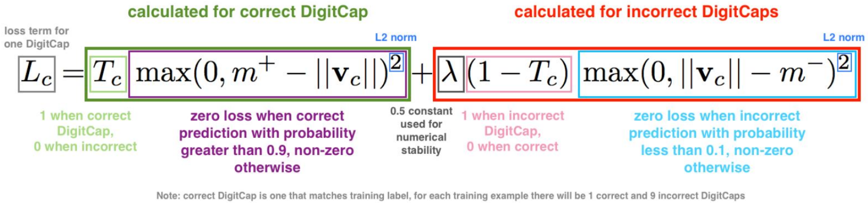
Each capsule computes a weighted average of the transformed input vectors in $\hat{\mathbf{u}}$. The "coupling" weights, $\mathbf{c}$, are determined on each forward pass via an iterative routing algorithm that acts as a sort of orientation-popularity filter. If multiple large vectors point in the same direction, they will get a large weight. Shorter vectors pointing in scattered directions will get a small weight.

The routing algorithm also uses softmax so that each of the 1152 input vectors in $\mathbf{u}$, sends most of its activity to just one of the ten outputs. The routing for $\mathbf{u}_{1151}$ is shown in red. Although both $\hat{\mathbf{u}}_{1151,0}$ and $\hat{\mathbf{u}}_{1151,1}$ have popular orientations in their respective digit capsules, only one has a large weight, $\mathbf{c}_{1151,0}$, so most of the activity flows to the digit 0 output.

The output vector is squashed so that its length can model the probability that the capsule's digit is present.

Mike Ross
mikenon@gmail.com

Loss Function



calculated for correct DigitCap

calculated for incorrect DigitCaps

loss term for one DigitCap

L2 norm

L2 norm

$$\boxed{L_c} = \boxed{\boxed{T_c}\,\boxed{\max(0, m^+ - ||\mathbf{v}_c||)}^{\boxed{2}}} + \boxed{\boxed{\lambda}\,\boxed{(1 - T_c)}\,\boxed{\max(0, ||\mathbf{v}_c|| - m^-)^{\boxed{2}}}}$$

1 when correct DigitCap, 0 when incorrect

zero loss when correct prediction with probability greater than 0.9, non-zero otherwise

0.5 constant used for numerical stability

1 when incorrect DigitCap, 0 when correct

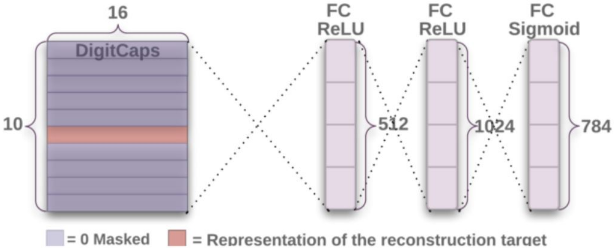
zero loss when incorrect prediction with probability less than 0.1, non-zero otherwise

Note: correct DigitCap is one that matches training label, for each training example there will be 1 correct and 9 incorrect DigitCaps

Outline

1. Capsule Network Intuition
   - Capsule Network - Intuition
   - Better Representation of Images: Inverse Graphics
2. Capsules
   - What are capsules
   - Capsule vs Neurons
   - Properties of capsules
3. Capsule Network
   - Capsule Network Encoder Architecture
   - Capsule Network Decoder Architecture
4. Experiments and Results
   - MNIST
   - multiMNIST
   - CIFAR10

Capsule Network Decoder Architecture



- reconstruction loss accounted for by sum of squares between original and predicted intensities

Outline

1. Capsule Network Intuition
   - Capsule Network - Intuition
   - Better Representation of Images: Inverse Graphics
2. Capsules
   - What are capsules
   - Capsule vs Neurons
   - Properties of capsules
3. Capsule Network
   - Capsule Network Encoder Architecture
   - Capsule Network Decoder Architecture
4. Experiments and Results
   - MNIST
   - multiMNIST
   - CIFAR10

Capsules in MNIST and Robustness to Affinity Transformations

- test errors on 28 x 28 MNIST with 60K and 10K training and testing datasets (with natural variance in skew, rotation, style, etc):

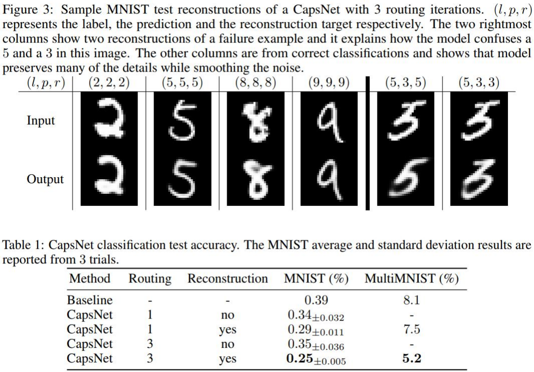


Figure 3: Sample MNIST test reconstructions of a CapsNet with 3 routing iterations. (l, p, r) represents the label, the prediction and the reconstruction target respectively. The two rightmost columns show two reconstructions of a failure example and it explains how the model confuses a 5 and a 3 in this image. The other columns are from correct classifications and shows that model preserves many of the details while smoothing the noise.

Table 1: CapsNet classification test accuracy. The MNIST average and standard deviation results are reported from 3 trials.

Method	Routing	Reconstruction	MNIST (%)	MultiMNIST (%)
Baseline	-	-	0.39	8.1
CapsNet	1	no	$0.34_{\pm 0.032}$	-
CapsNet	1	yes	$0.29_{\pm 0.011}$	7.5
CapsNet	3	no	$0.35_{\pm 0.036}$	-
CapsNet	3	yes	$\mathbf{0.25}_{\pm 0.005}$	**5.2**

Capsules in MNIST and Robustness to Affinity Transformations

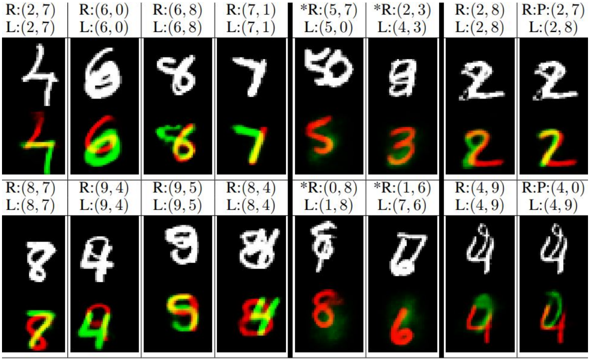
- tested against traditional CNN with maxpooling and dropout
- both trained with MNIST digit placed randomly on a black background of 40 40 px
- tested on the affNIST
- traditional CNN: 99.23% accuracy on the expanded MNIST test set achieved 79% accuracy on the affNIST test set
- Capsule Network with similar parameters: similar accuracy (99.22%) on the expanded mnist test set only achieved 66% on the affNIST test set.

Outline

1. Capsule Network Intuition
   - Capsule Network - Intuition
   - Better Representation of Images: Inverse Graphics
2. Capsules
   - What are capsules
   - Capsule vs Neurons
   - Properties of capsules
3. Capsule Network
   - Capsule Network Encoder Architecture
   - Capsule Network Decoder Architecture
4. Experiments and Results
   - MNIST
   - multiMNIST
   - CIFAR10

multiMNIST - Segmenting highly overlapping digits

- Dynamic routing can be viewed as a parallel attention mechanism that allows each capsule at one level to attend to some active capsules at the level below and to ignore others
- trained from scratch on the multiMNIST database (Lower L: predicted)



multiMNIST - Results

- Network ability to reconstruct digits regardless of the overlap shows that each digit capsule can pick up the style and position from the votes it is receiving from PrimaryCapsules layer.

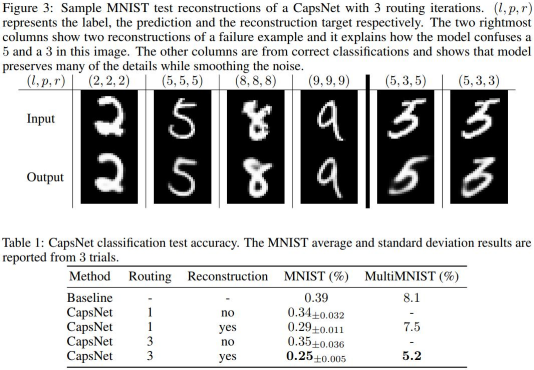


Figure 3: Sample MNIST test reconstructions of a CapsNet with 3 routing iterations. (l, p, r) represents the label, the prediction and the reconstruction target respectively. The two rightmost columns show two reconstructions of a failure example and it explains how the model confuses a 5 and a 3 in this image. The other columns are from correct classifications and shows that model preserves many of the details while smoothing the noise.

Table 1: CapsNet classification test accuracy. The MNIST average and standard deviation results are reported from 3 trials.

Method	Routing	Reconstruction	MNIST (%)	MultiMNIST (%)
Baseline	-	-	0.39	8.1
CapsNet	1	no	$0.34_{\pm 0.032}$	-
CapsNet	1	yes	$0.29_{\pm 0.011}$	7.5
CapsNet	3	no	$0.35_{\pm 0.036}$	-
CapsNet	3	yes	$\mathbf{0.25}_{\pm 0.005}$	**5.2**

Outline

1. Capsule Network Intuition
   - Capsule Network - Intuition
   - Better Representation of Images: Inverse Graphics
2. Capsules
   - What are capsules
   - Capsule vs Neurons
   - Properties of capsules
3. Capsule Network
   - Capsule Network Encoder Architecture
   - Capsule Network Decoder Architecture
4. Experiments and Results
   - MNIST
   - multiMNIST
   - CIFAR10

CIFAR10

- CIFAR10 and achieved 10.6% error with an ensemble of 7 models each of which is trained with 3 routing iterations
    - standard CNN achieves 10.6% error
    - Each model has the same architecture as the simple model we used for MNIST except that there are three color channels and we used 64 different types of primary capsule
    - introduced a ”none-of-the-above” category for the routing softmaxes, since we do not expect the final layer of ten capsules to explain everything in the image
- Drawback: tends to account for everything in the image so it does better when it can model the clutter than when it just uses an additional orphan category in the dynamic routing.
    - In CIFAR-10, the backgrounds are much too varied to model in a reasonable sized net which helps to account for the poorer performance.

Thank you for Listening

For Further Reading I

Sara Sabour, Nicholas Frosst Geoffrey E. Hinton.
Dynamic Routing Between Capsules.
Google Brain, 2017.

S. Someone.
On this and that.
Journal of This and That, 2(1):50–100, 2000.