Capsule Networks

Speaker : Semin Choi

Department of Statistics, Seoul National University, South Korea

November 29, 2017
The Drawbacks of CNNs

- Let’s consider a very simple and non-technical example:

To a CNN, both pictures are similar, since they both contain similar elements.

- Why?
The Drawbacks of CNNs

- The key idea of CNNs: Increasing the “field of view” of higher layer’s neurons.
  - Max-pooling, Successive convolutional layers
  - “Invariance”
  - Information loss

- Hinton stated that the fact that max pooling is working so well is a big mistake and a disaster.

- Internal data representation of a CNN does not take into account important spatial hierarchies between simple and complex objects.
Inverse Graphics Approach

Rendering:

<table>
<thead>
<tr>
<th>Rectangle</th>
<th>Triangle</th>
</tr>
</thead>
<tbody>
<tr>
<td>x=20</td>
<td>x=24</td>
</tr>
<tr>
<td>y=30</td>
<td>y=25</td>
</tr>
<tr>
<td>angle=16°</td>
<td>angle=-65°</td>
</tr>
</tbody>
</table>

Instantiation parameters → Rendering → Image

Inverse Graphics:

<table>
<thead>
<tr>
<th>Rectangle</th>
<th>Triangle</th>
</tr>
</thead>
<tbody>
<tr>
<td>x=20</td>
<td>x=24</td>
</tr>
<tr>
<td>y=30</td>
<td>y=25</td>
</tr>
<tr>
<td>angle=16°</td>
<td>angle=-65°</td>
</tr>
</tbody>
</table>

Instantiation parameters ← Inverse rendering ← Image
Inverse Graphics Approach

- Your brain can easily recognize this is the same object.
- CNNs do not have this capability.
Capsules

- Hinton et al. (2011)

- A capsule is a group of neurons whose activity vector represents the instantiation parameters of a specific type of entity such as an object or an object part.

**Activation vector:**
- **Length** = estimated probability of presence
- **Orientation** = object’s estimated pose parameters
Equivariance
A hierarchy of Capsules

- **Boat**
  - Rectangle
    - x=20
    - y=30
    - angle=16°
  - Triangle
    - x=24
    - y=25
    - angle=-65°

- **House**
  - Rectangle
    - x=20
    - y=30
    - angle=-5°
  - Triangle
    - x=26
    - y=31
    - angle=137°
How the vector inputs and outputs of a capsule are computed

- Input: $u_i$, Output: $s_j$

$$\hat{u}_{j|i} = W_{ij}u_i, \quad s_j = \sum_i c_{ij} \hat{u}_{j|i},$$

where the $c_{ij}$ are coupling coefficients that are determined by the iterative dynamic routing process.

$$c_{ij} = \frac{\exp(b_{ij})}{\sum_k \exp(b_{ik})}.$$

- Squashing Function:

$$v_j = \frac{\|s_j\|^2}{1 + \|s_j\|^2} \frac{s_j}{\|s_j\|}$$
How the vector inputs and outputs of a capsule are computed

**Procedure 1** Routing algorithm.

1. **procedure** ROUTING($\hat{u}_{j|i}$, $r$, $l$)
2. for all capsule $i$ in layer $l$ and capsule $j$ in layer $(l + 1)$: $b_{ij} \leftarrow 0$.
3. for $r$ iterations do
4.   for all capsule $i$ in layer $l$: $c_i \leftarrow \text{softmax}(b_i)$  \hspace{1cm} ▷ softmax computes Eq. 3
5.   for all capsule $j$ in layer $(l + 1)$: $s_j \leftarrow \sum_i c_{ij} \hat{u}_{j|i}$
6.   for all capsule $j$ in layer $(l + 1)$: $v_j \leftarrow \text{squash}(s_j)$  \hspace{1cm} ▷ squash computes Eq. 1
7. for all capsule $i$ in layer $l$ and capsule $j$ in layer $(l + 1)$: $b_{ij} \leftarrow b_{ij} + \hat{u}_{j|i}.v_j$

return $v_j$
How the vector inputs and outputs of a capsule are computed
How the vector inputs and outputs of a capsule are computed

**Strong agreement!**

The rectangle and triangle capsules should be routed to the boat capsules.

\[ s_j = \text{weighted sum} \]
\[ v_j = \text{squash}(s_j) \]

Predicted Outputs

Primary Capsules

Actual outputs of the next layer capsules (round #1)
How the vector inputs and outputs of a capsule are computed
How the vector inputs and outputs of a capsule are computed
Margin loss for class existence:

\[ L_k = T_k \max(0, m^+ - \|v_k\|^2) + \lambda(1 - T_k) \max(0, \|v_k\| - m^-)^2 \]

where \( T_k = 1 \) iff a class \( k \) is present.
CapsNet

- A simple CapsNet for MNIST data

\[ W_{ij} = [8 \times 16] \]

- Loss = Margin loss + \( \alpha \) reconstruction loss
MultiMNIST data