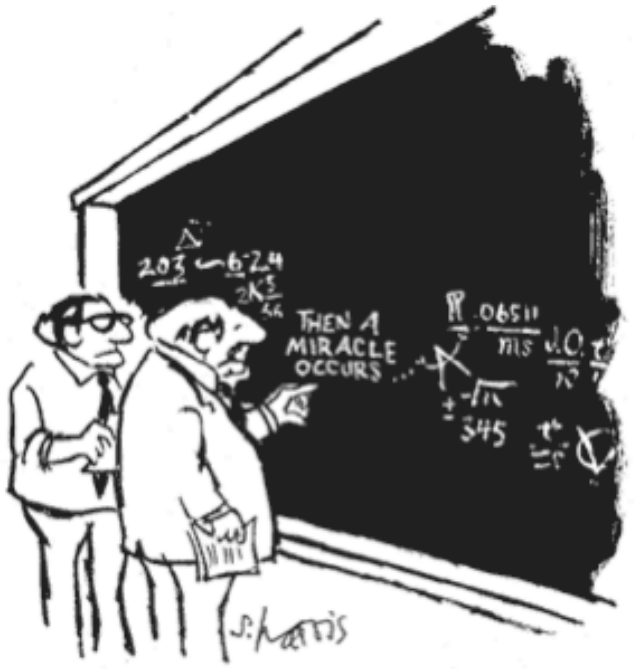
Getting inside a CNN

Morten Nielsen
Department of Health Technology,
DTU

Why do we need to get inside?

- Current ML library functions (in TensorFlow/Pytorch etc) have limited flexibility
- Gaining access to the information stored in the CNN is a non-trivial exercise
- My entire carrier is build on getting inside a computational model and fine tuning it to archive improved performance for receptor-ligand (read peptide-MHC) systems
- I hate using a black boxes



"I THINK YOU SHOULD BE MORE EXPLICIT HERE IN STEP TWO."

Deep(er) Network architecture

$$E = \tfrac{1}{2} \cdot (O - t)^2$$

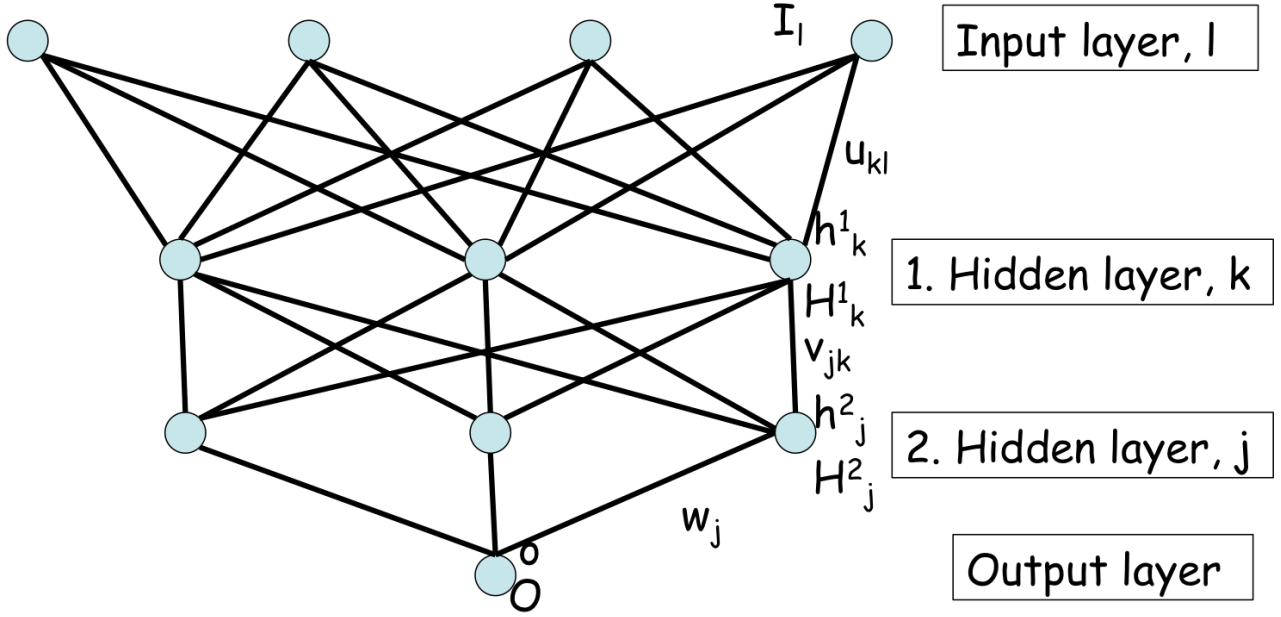
$$O = g(o), H = g(h)$$

$$g(x) = \frac{1}{1 + e^{-x}}$$

$$o = \sum_j w_j \cdot H^2_j$$

$$h^2_j = \sum_k v_{jk} \cdot H^1_k$$

$$h^1_k = \sum_l u_{kl} \cdot I_l$$



$$\Delta w_i = -\varepsilon \cdot \frac{\partial E}{\partial w_i}$$

$$O = g(o)$$

$$g(x) = \frac{1}{1 + e^{-x}}$$

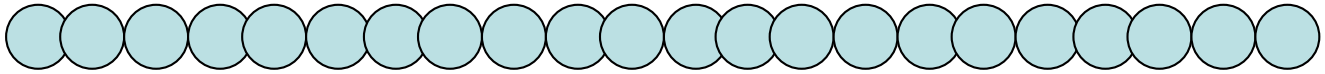
$$g'(x) = \frac{-1}{(1 + e^{-x})^2} \cdot (-e^{-x})$$

$$= (1 - g(x)) \cdot g(x)$$

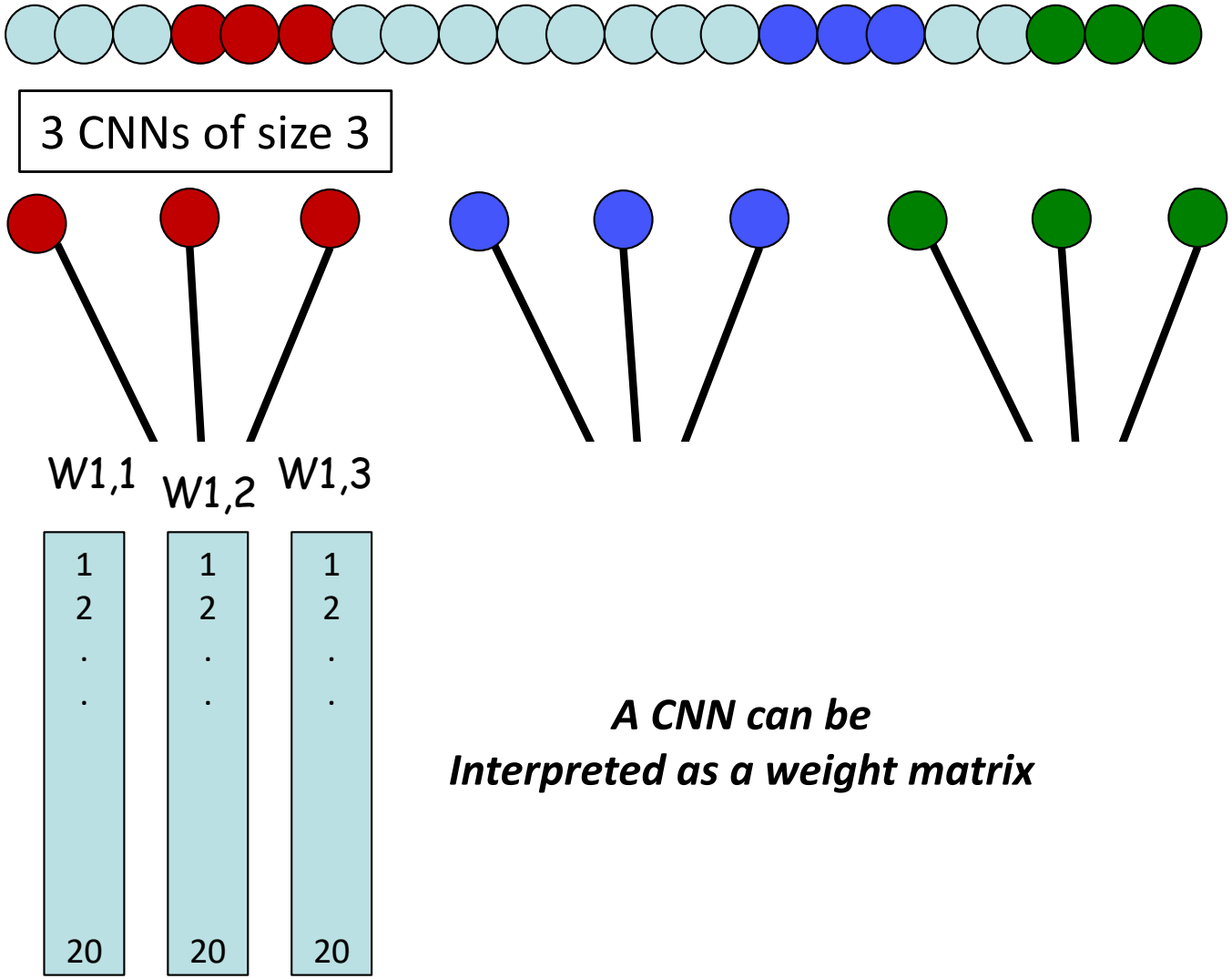
$$g'(o) = \big(1 - g(o)\big) * g(o)$$

$$= (1 - O) * O$$

Making a (max-pooled) CNN

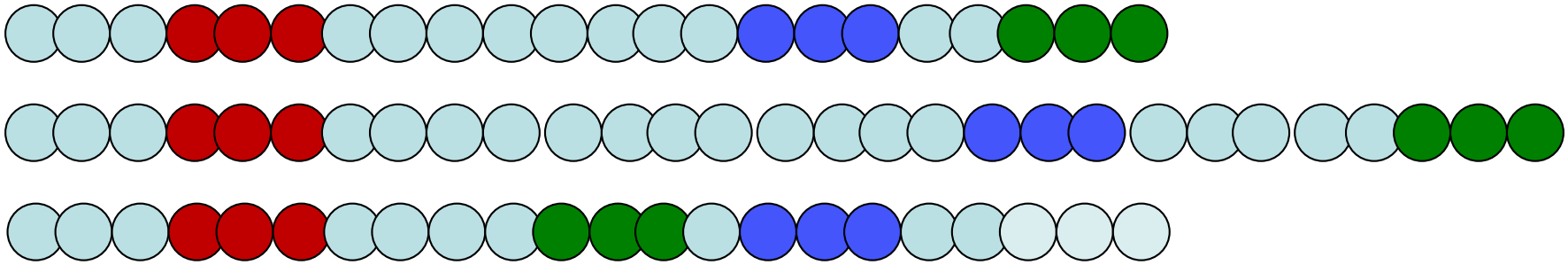


Making a (max-pooled) CNN



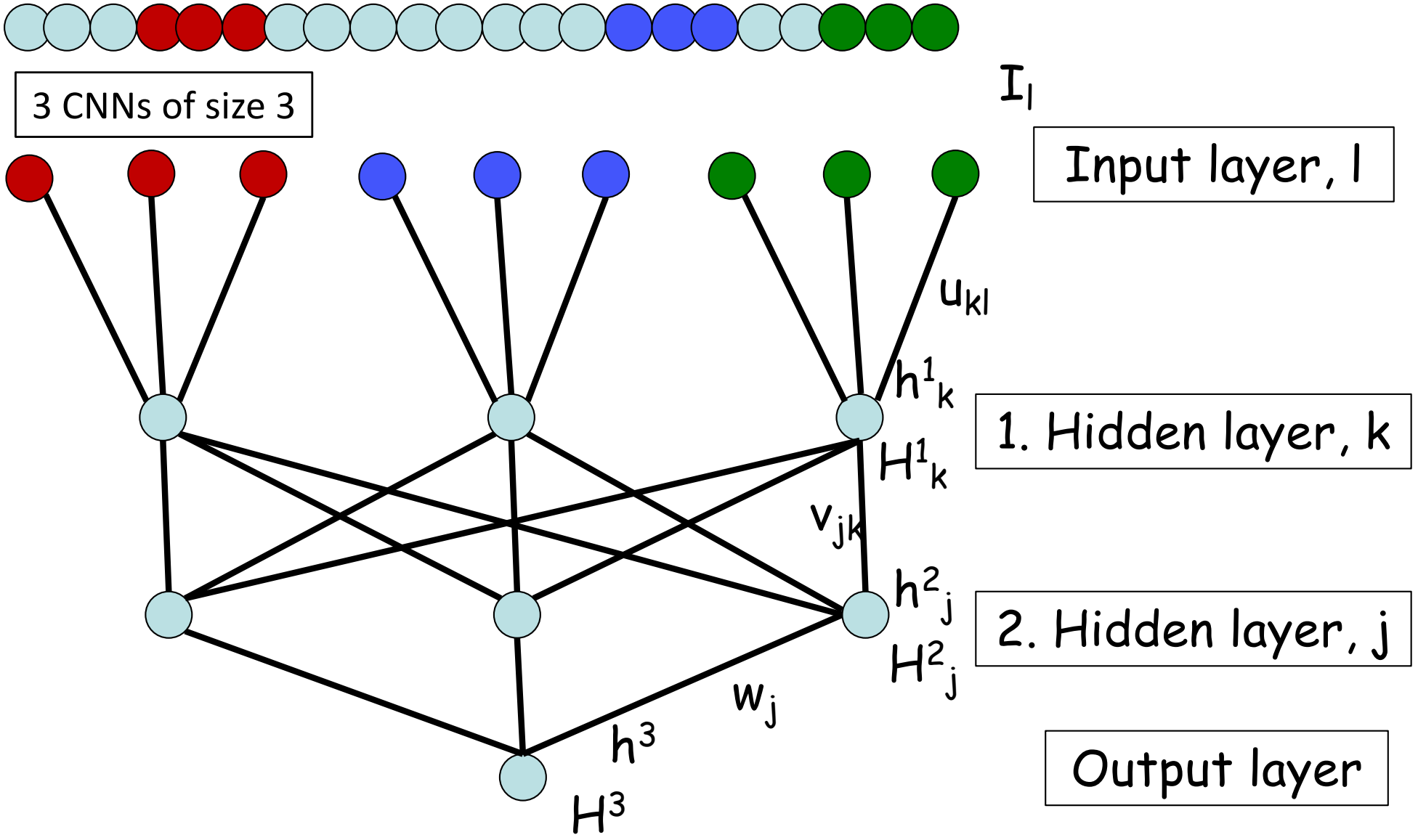
A CNN can be
Interpreted as a weight matrix

Making a (max-pooled) CNN

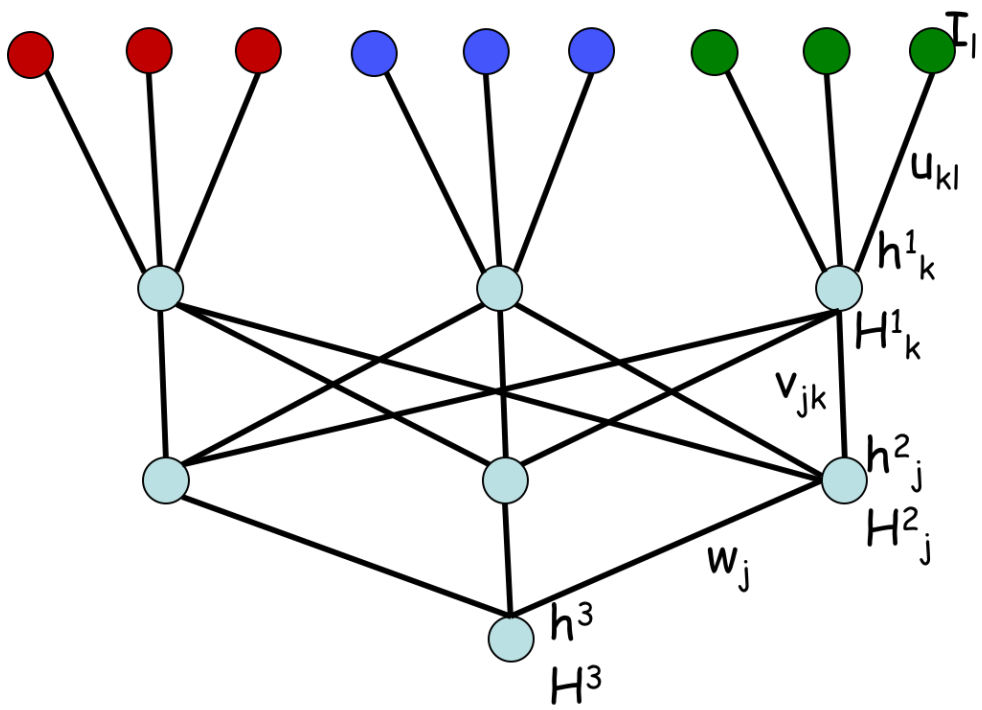


Handling input of variable length (and potential invertions)
This is not trivially possible using FFNN

Making a (max-pooled) CNN



Network architecture (hidden to hidden)



Input layer, l

1. Hidden layer, k

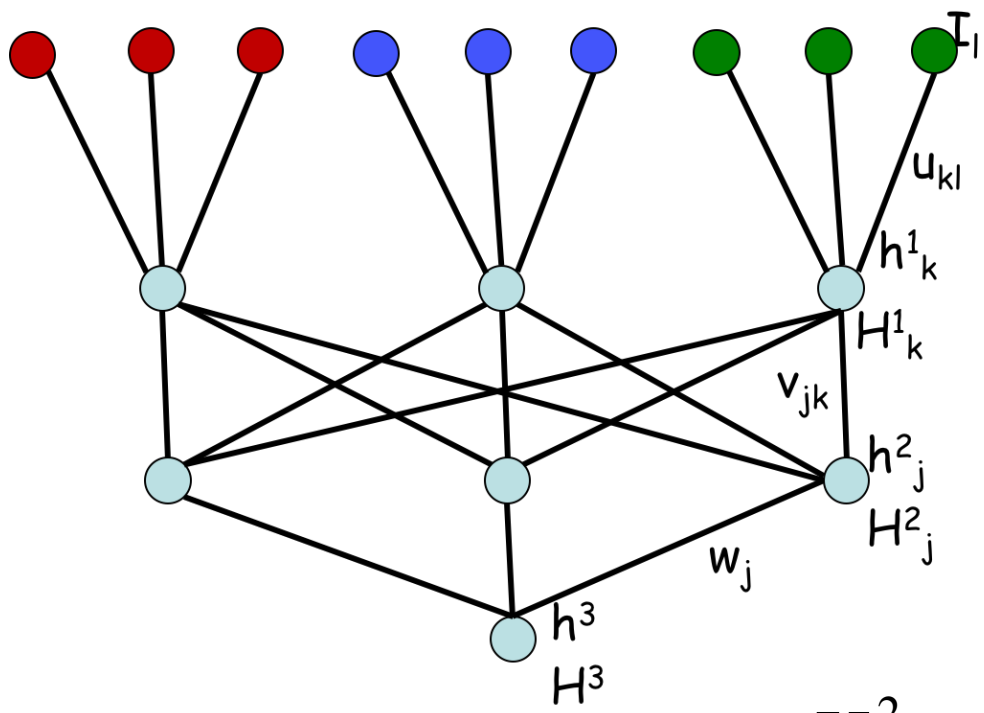
2. Hidden layer, j

Output layer

$$\frac{\partial E}{\partial w_j} = \frac{\partial E(H^3(h^3(w_j)))}{\partial w_j} = \frac{\partial E}{\partial H^3} \cdot \frac{\partial H^3}{\partial h^3} \cdot \frac{\partial h^3}{\partial w_j} = \underbrace{(H^3 - t) \cdot g'(h^3)}_{\delta_3} \cdot H_j^2$$

$$= \delta_3 * H_j^2$$

Network architecture (hidden to hidden)



Input layer, l

1. Hidden layer, k

2. Hidden layer, j

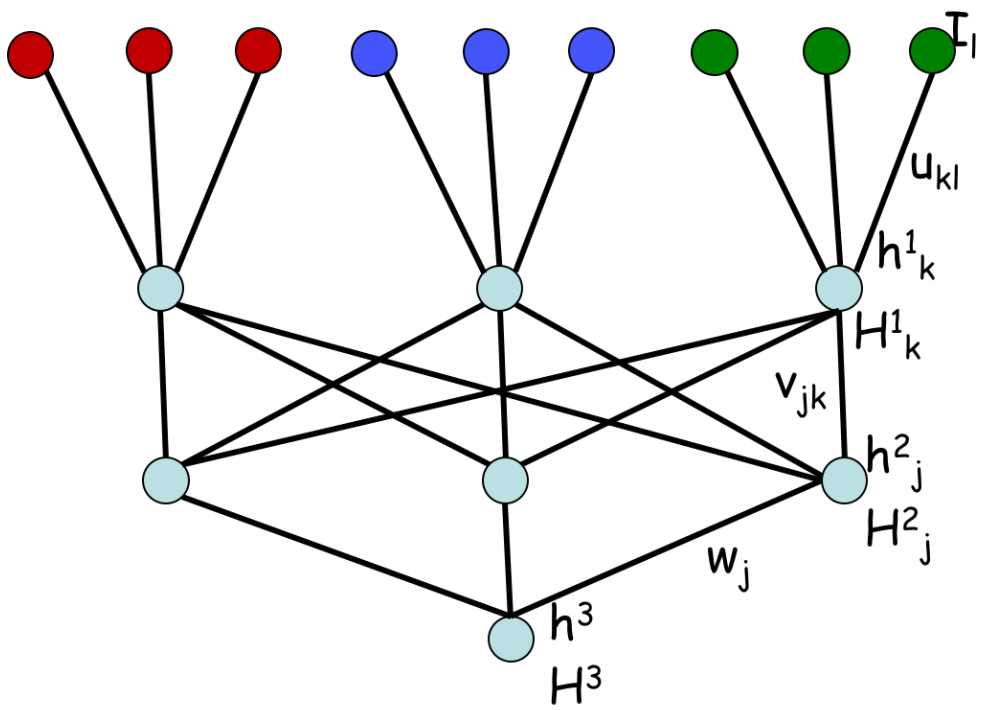
Output layer

$$\frac{\partial E}{\partial v_{jk}} = \frac{\partial E}{\partial H^3} \cdot \frac{\partial H^3}{\partial h^3} \cdot \frac{\partial h^3}{\partial H^2_j} \cdot \frac{\partial H^2_j}{\partial h^2_j} \cdot \frac{\partial h^2_j}{\partial v_{jk}}$$

$$= \underbrace{(H^3 - t) \cdot g'(h^3)}_{\boldsymbol{\delta_3}} \cdot w_j \cdot g'(h^2_j) \cdot H^1_k$$

$$\delta_2^j = g'\left(h_j^2\right) * \delta_3 * w_j$$

Network architecture (input to hidden)



Input layer, l

1. Hidden layer, k

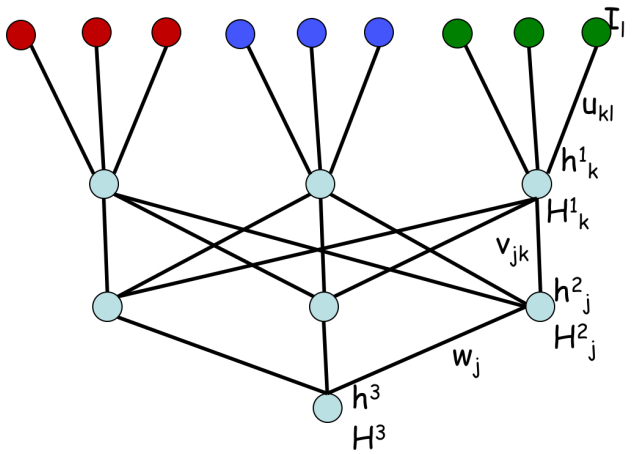
2. Hidden layer, j

Output layer

$$\frac{\partial E}{\partial u_{kl}} = \frac{\partial E}{\partial H^3} \cdot \frac{\partial H^3}{\partial h^3} \cdot \sum_j \frac{\partial h^3}{\partial H^2_j} \cdot \frac{\partial H^2_j}{\partial h^2_j} \cdot \frac{\partial h^2_j}{\partial H^1_k} \cdot \frac{\partial H^1_k}{\partial h^1_k} \cdot \frac{\partial h^1_k}{\partial u_{kl}}$$

$$= (H^3 - t) \cdot g'(h^3) \cdot \sum_j w_j \cdot g'(h^2_j) \cdot v_{jk} \cdot g'(h^1_k) \cdot I_l$$

The final eqautions



$$\frac{\partial E}{\partial w_{ji}^q} = \frac{\partial E}{\partial h_j^q} \cdot \frac{\partial h_j^q}{\partial w_{ji}^q} = \delta_j^q \cdot H_i^{q-1}$$

$$\delta_j^q = \frac{\partial E}{\partial h_j^q}$$

$$\delta^3 = \frac{\partial E}{\partial h^3} = \frac{\partial E}{\partial H^3} \cdot \frac{\partial H^3}{\partial h^3} = (H^3 - t) \cdot g'(h^3)$$

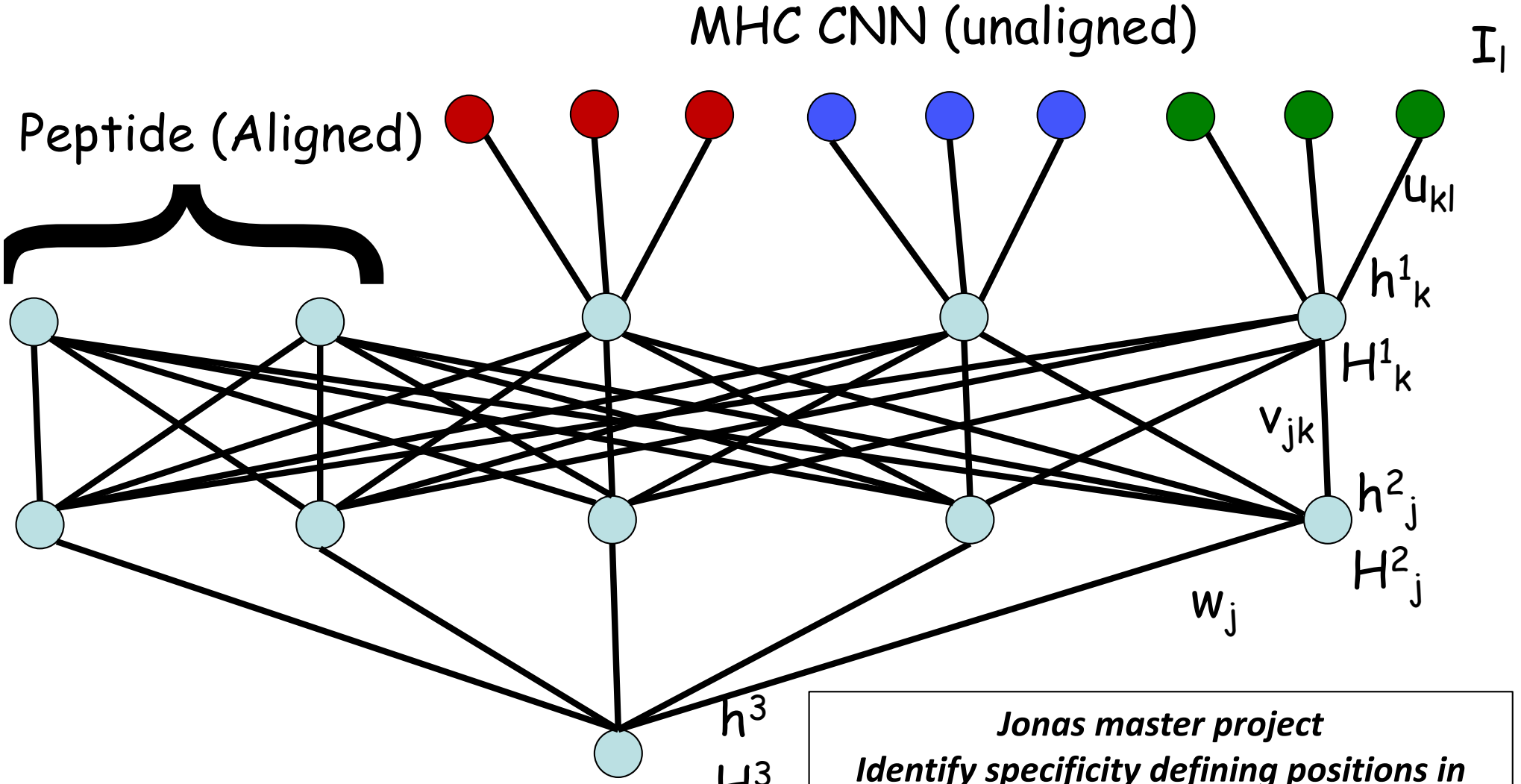
$$\delta_j^2 = \frac{\partial E}{\partial h_j^2} = \frac{\partial E}{\partial h^3} \cdot \frac{\partial h^3}{\partial h_j^2} = \frac{\partial E}{\partial h^3} \cdot \frac{\partial h^3}{\partial H_j^2} \cdot \frac{\partial H_j^2}{\partial h_j^2} = g'(h_j^2) \cdot \delta^3 \cdot v_{jk}$$

$$\delta_k^1 = \frac{\partial E}{\partial h_k^1} = \sum_j \frac{\partial E}{\partial h_j^2} \cdot \frac{\partial h_j^2}{\partial h_k^1} = \sum_j \frac{\partial E}{\partial h_j^2} \cdot \frac{\partial h_j^2}{\partial H_k^1} \cdot \frac{\partial H_k^1}{\partial h_k^1} = g'(h_k^1) \cdot \sum_j \delta_j^2 \cdot v_{jk}$$

$$h_j = \sum_i w_{ji} H_i$$

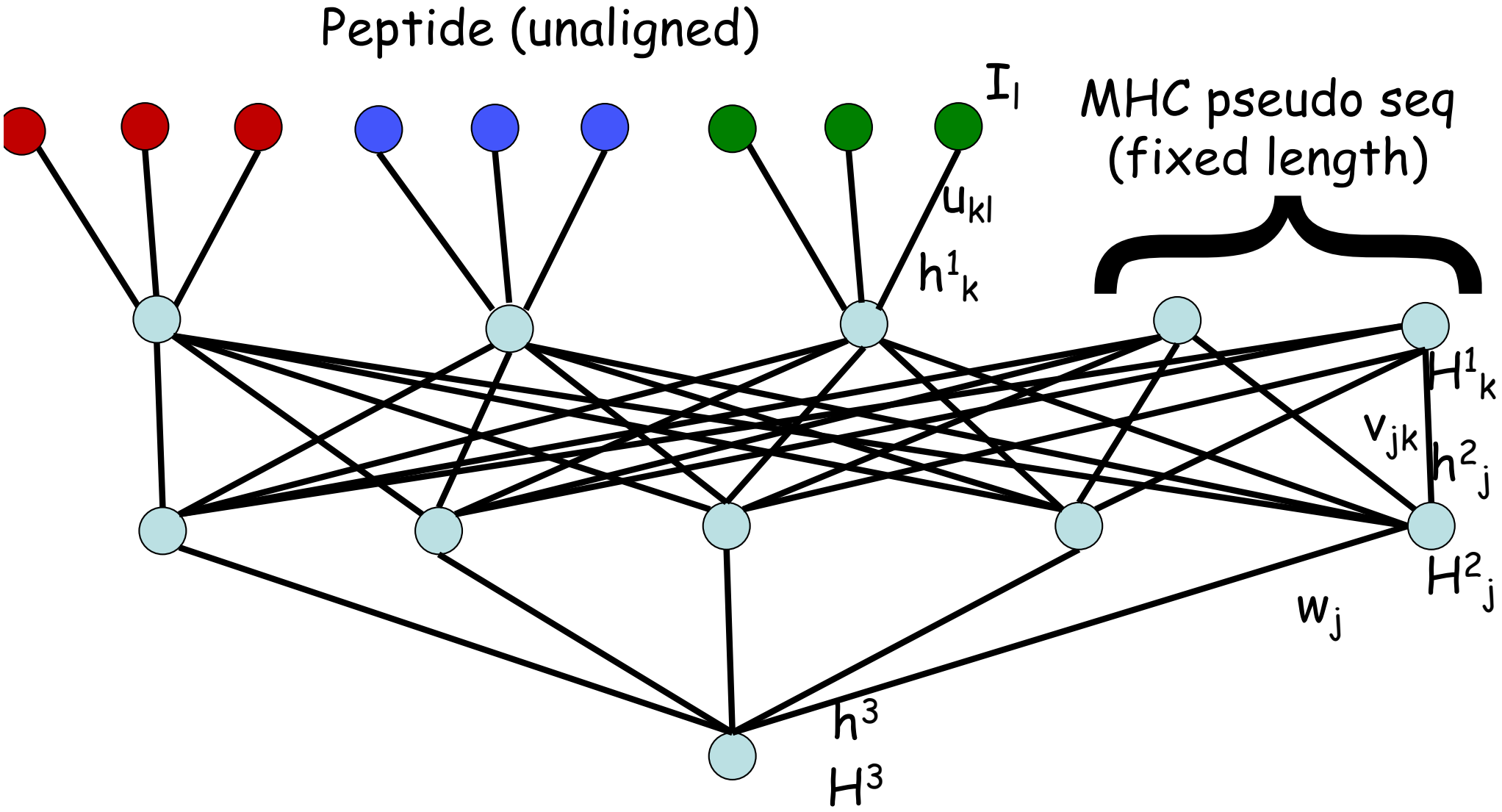
$$H_i = g(h_i)$$

Different architectures

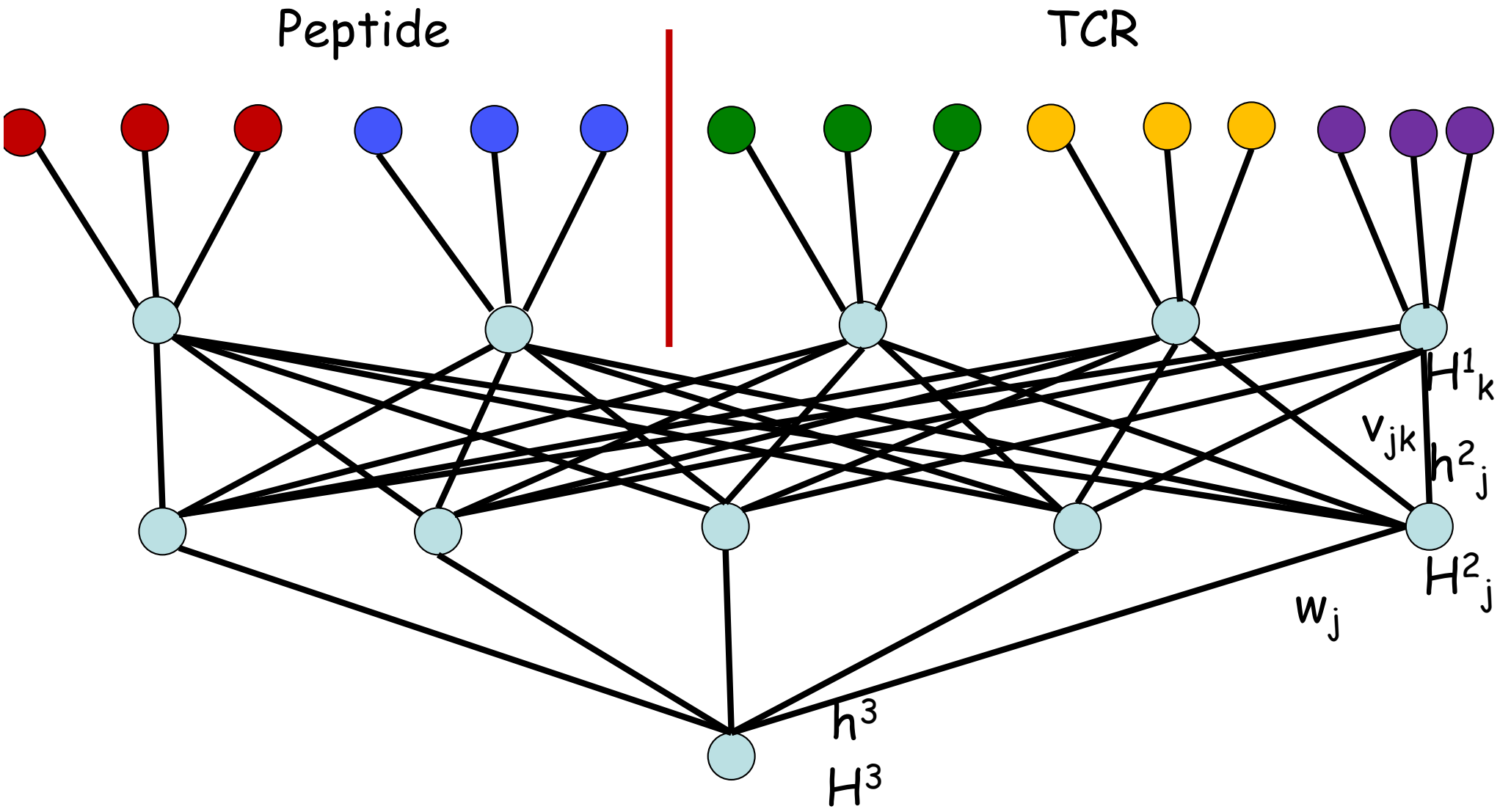


Jonas master project
Identify specificity defining positions in MHC sequences -> improve predictive power

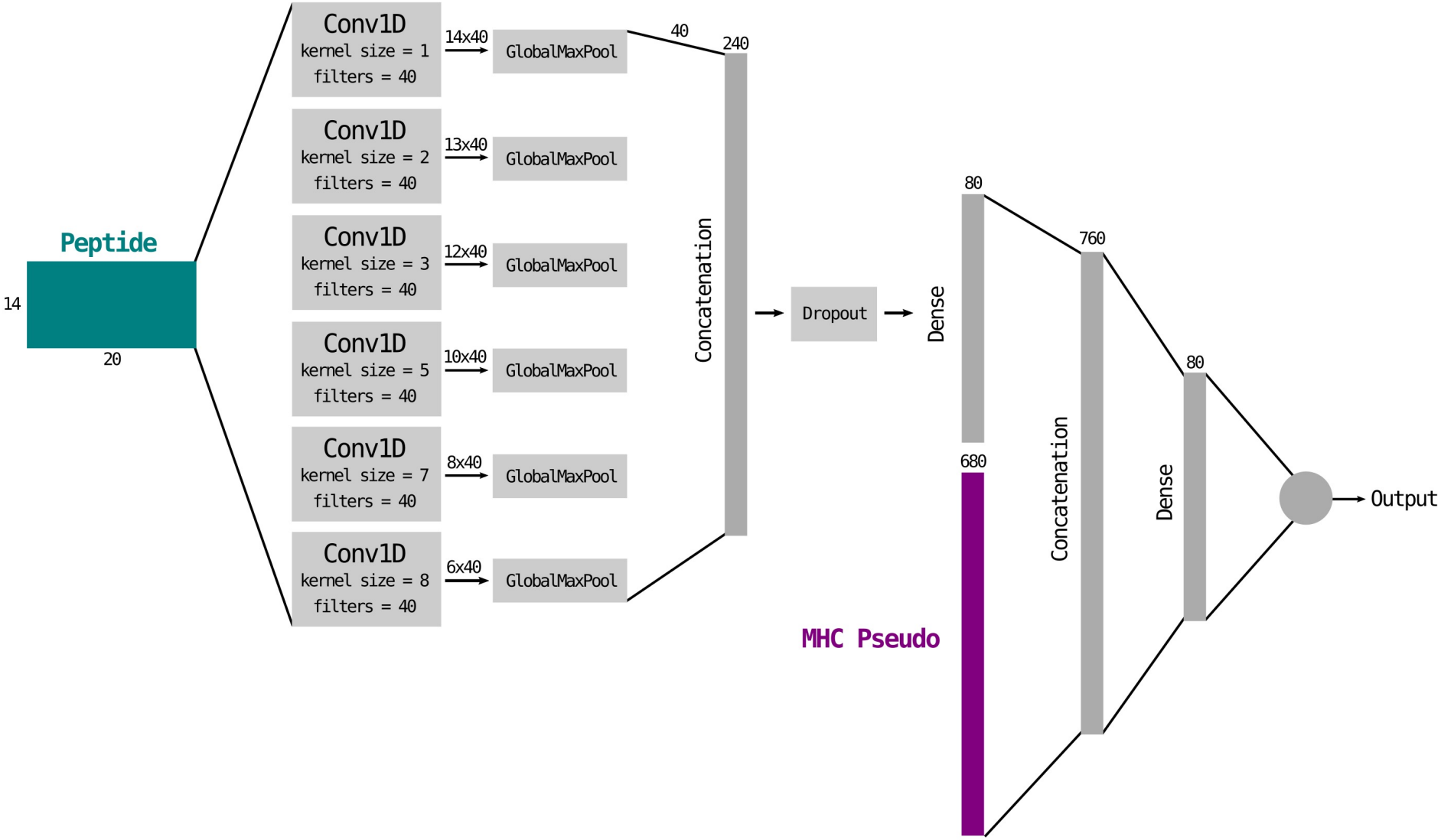
NetMHCpan/NetMHCIIpan as CNNs



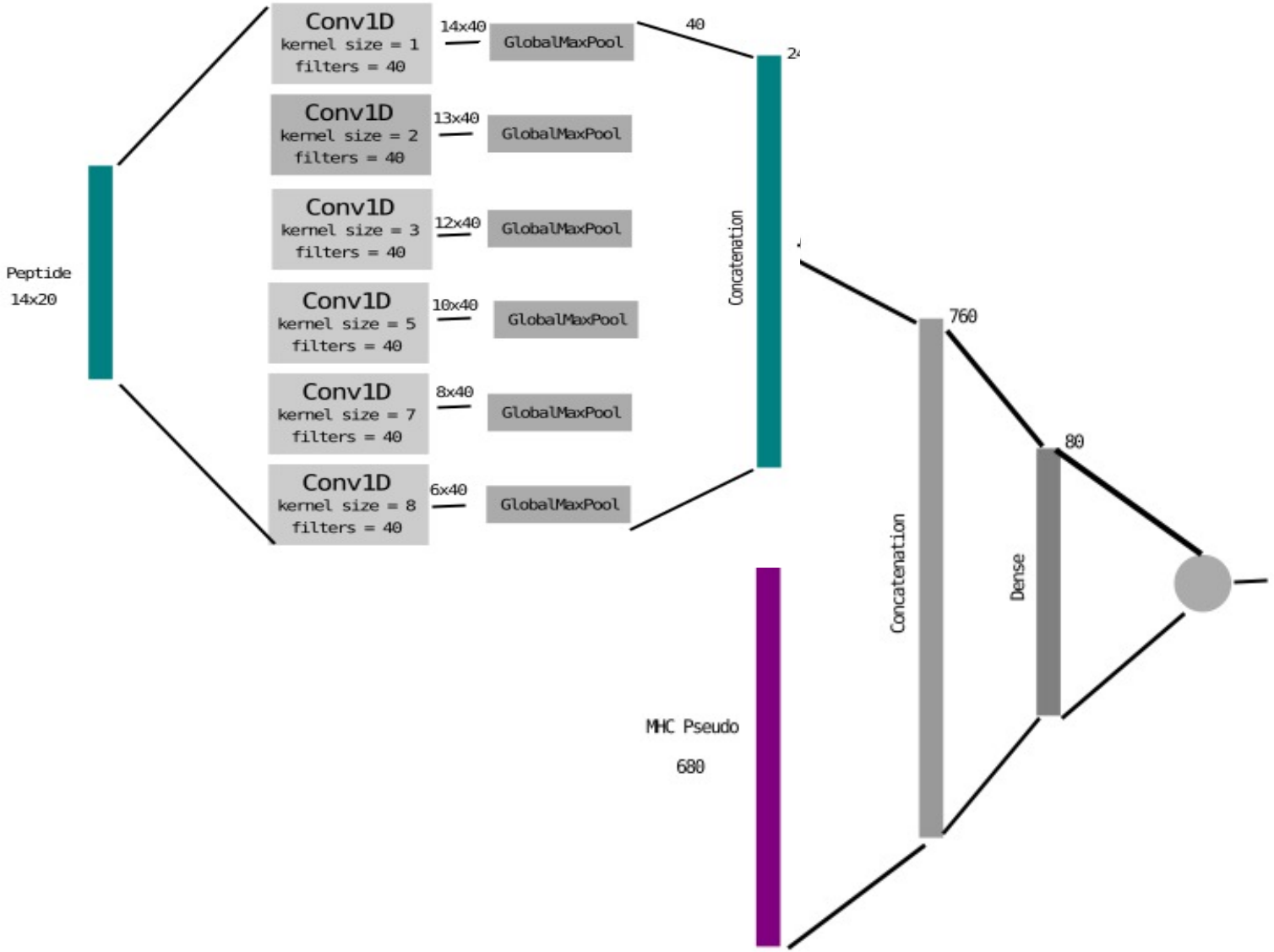
NetTCR



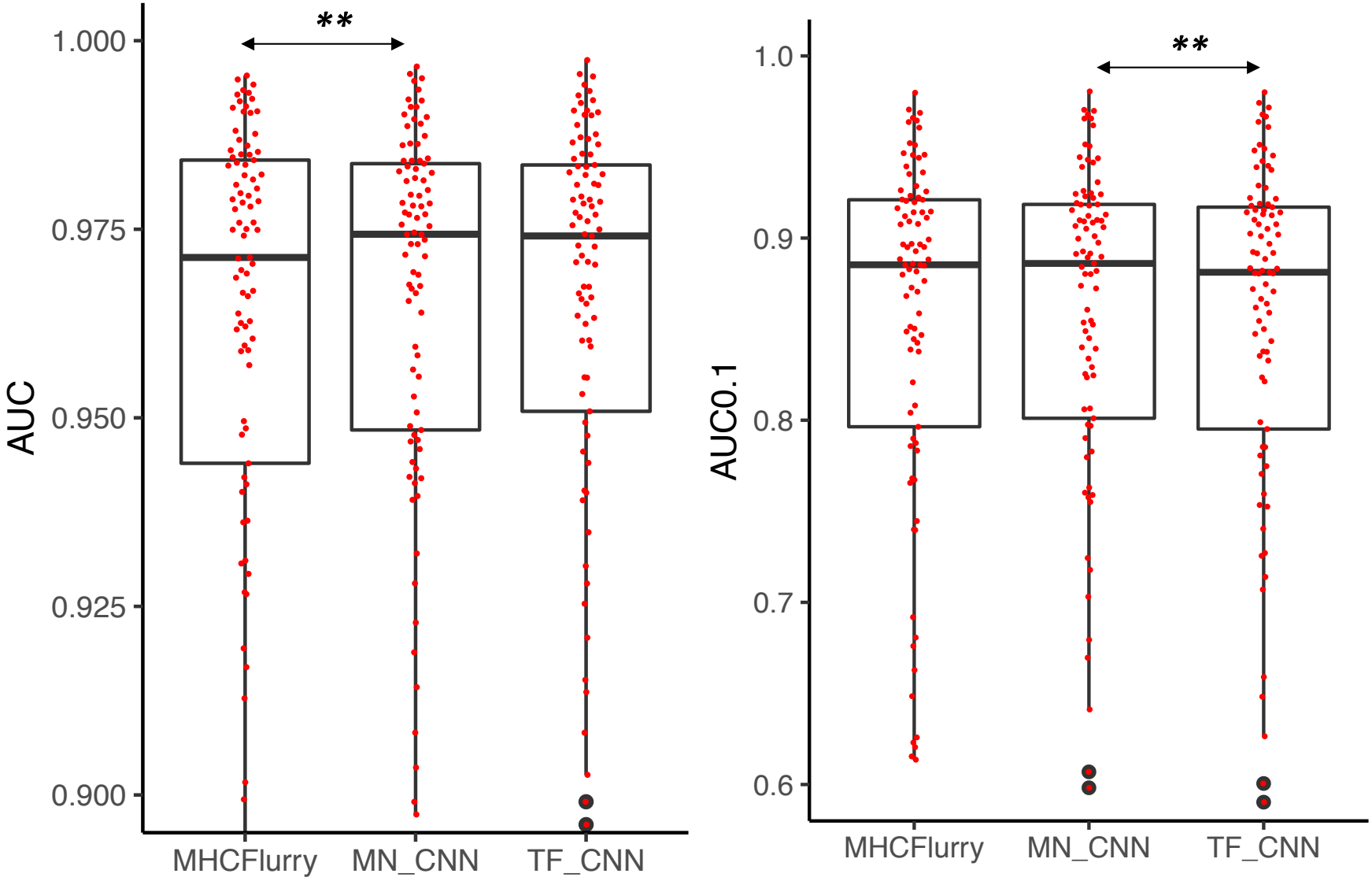
TF architecture (work by Bruno A)



MN architecture

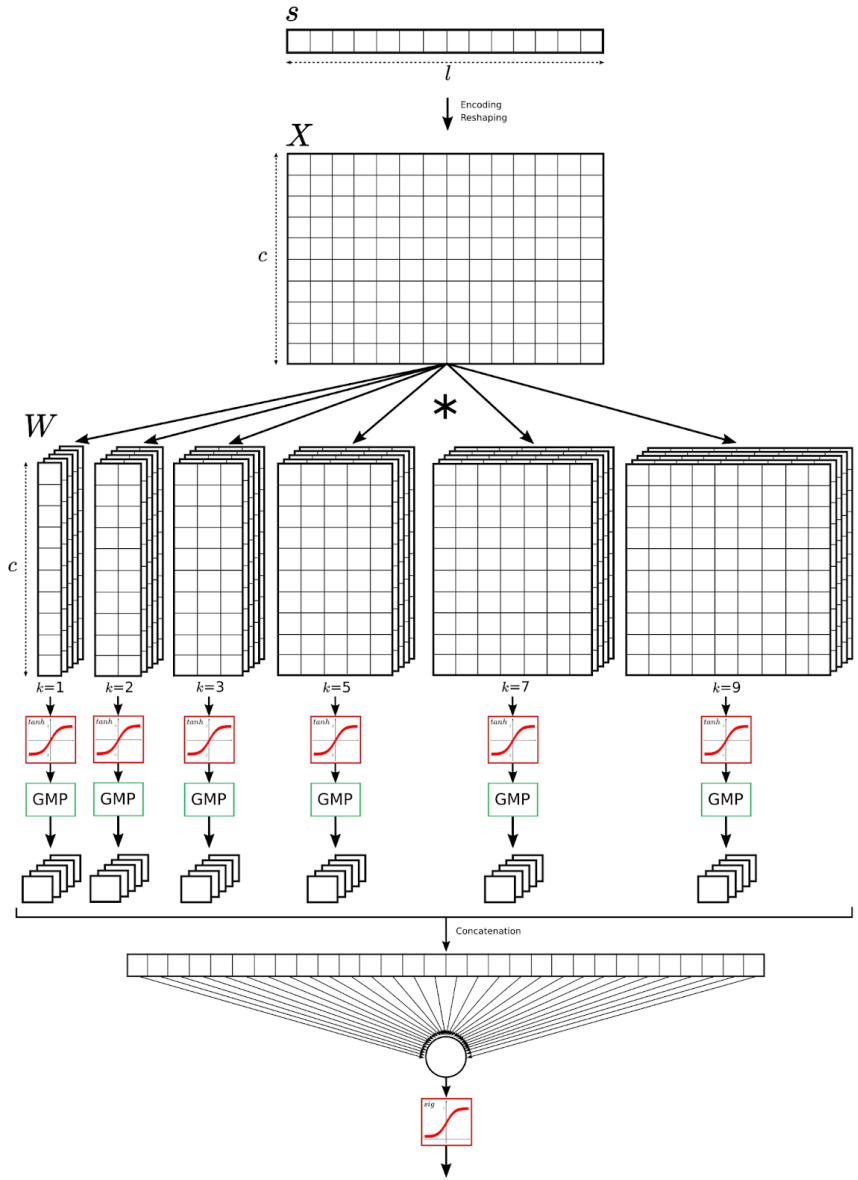


NetMHCpan-CNN

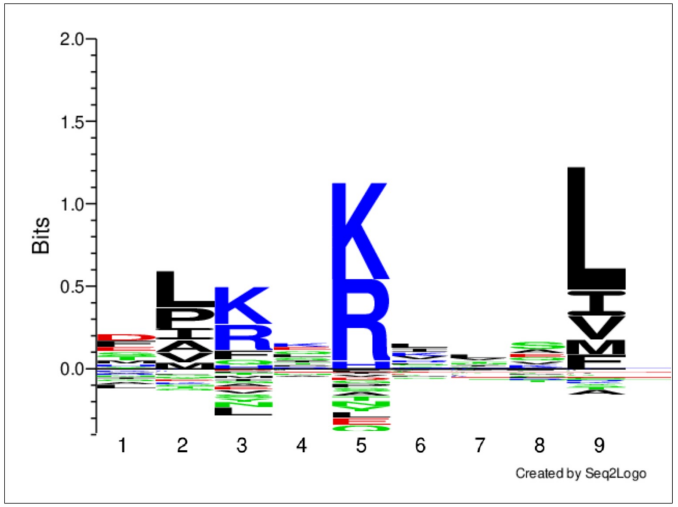


Getting inside a CNN
The first simple allele-specific architure

CENTERFO
RBIOLOGI
CALSEQU
ENCEANA
LYSIS CBS

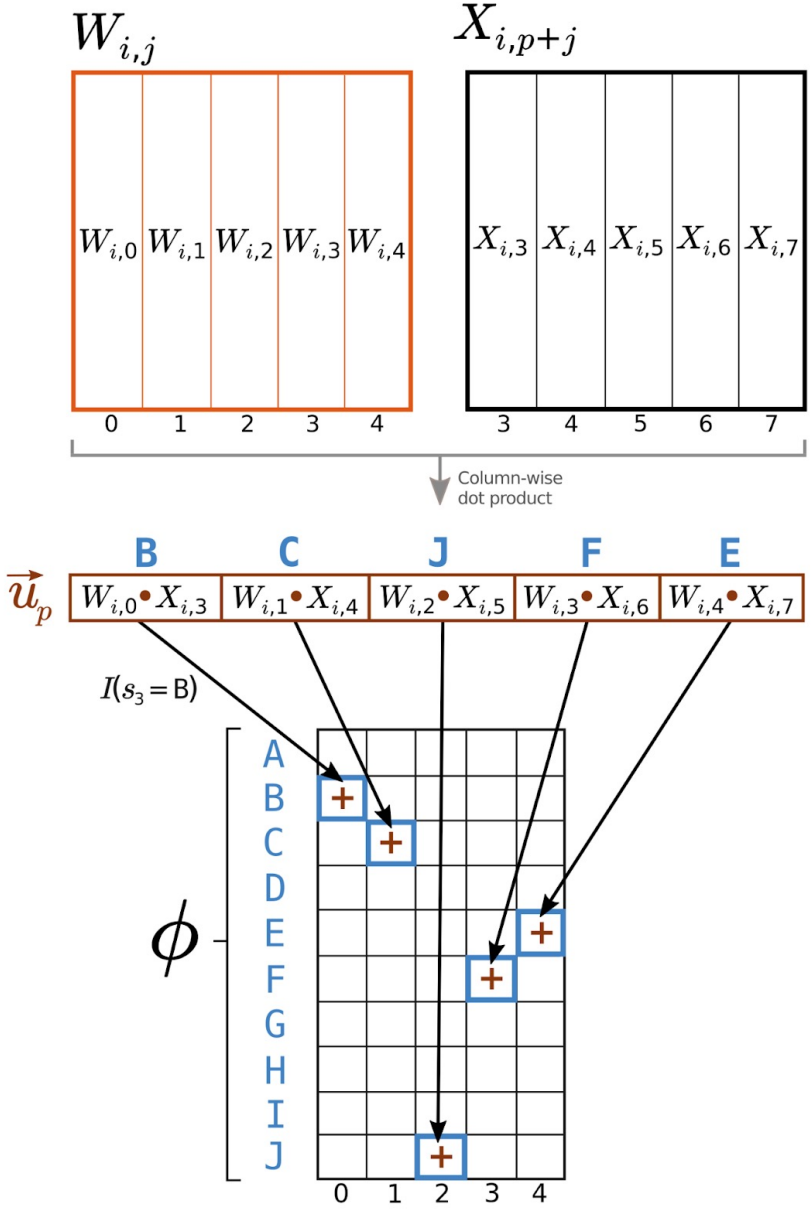


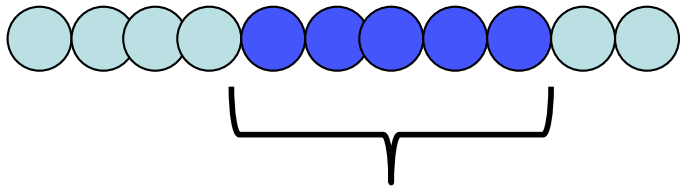
Motif learned by the model



Bruno Alverez, manuscript in preparation

Peptide projections

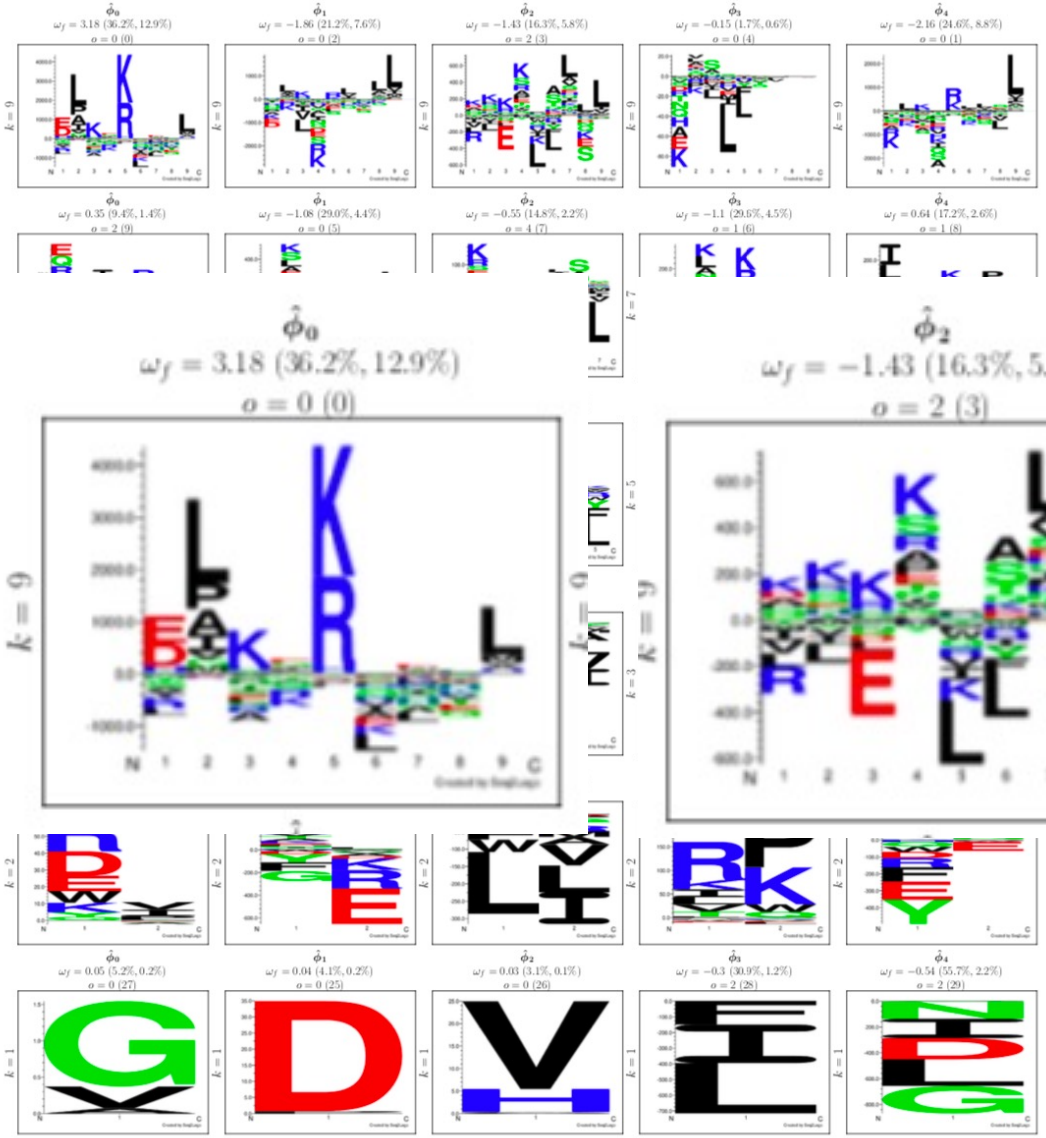


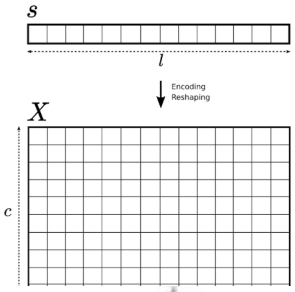


Max-pooled location

Bruno Alverez, manuscript in preparation

CNN filter motifs



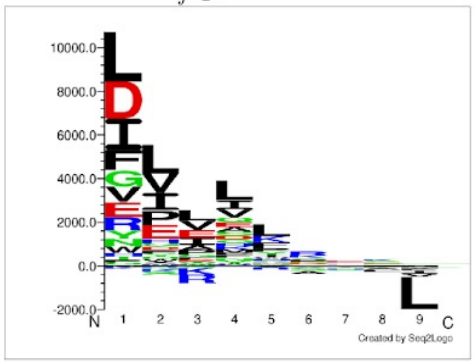


Bruno Alverez, manuscript in preparation

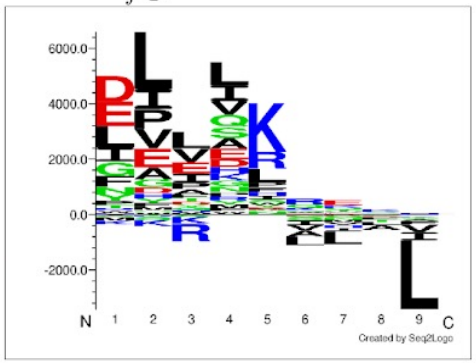
Motif reconstruction

CENTERFO
RBIOLOGI
CALSEQU
ENCEANA
LYSIS CBS

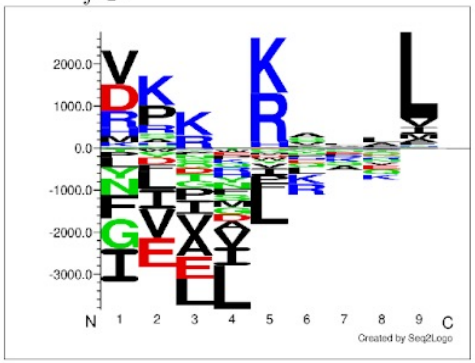
$\sum_{f \in F} \phi_f$



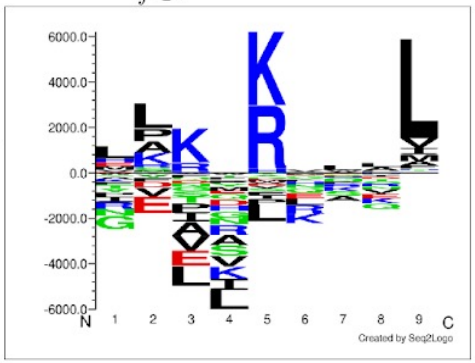
$\sum_{f \in F} |\omega_f| \cdot \phi_f$

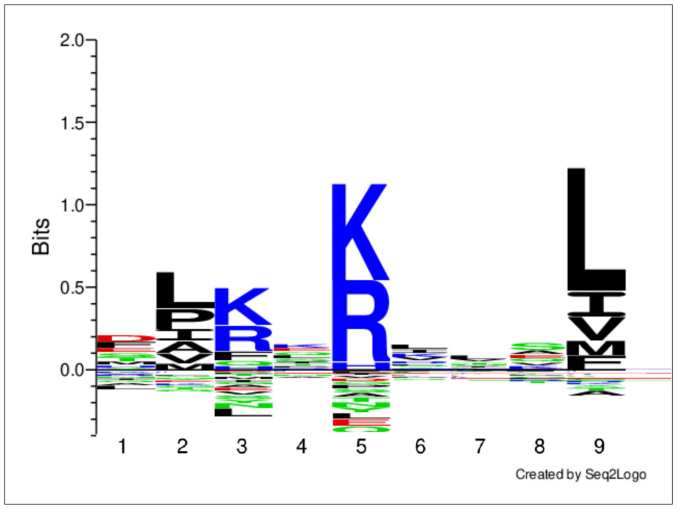


$\sum_{f \in F} sgn(\omega_f) \cdot \phi_f$



$\sum_{f \in F} \omega_f \cdot \phi_f$

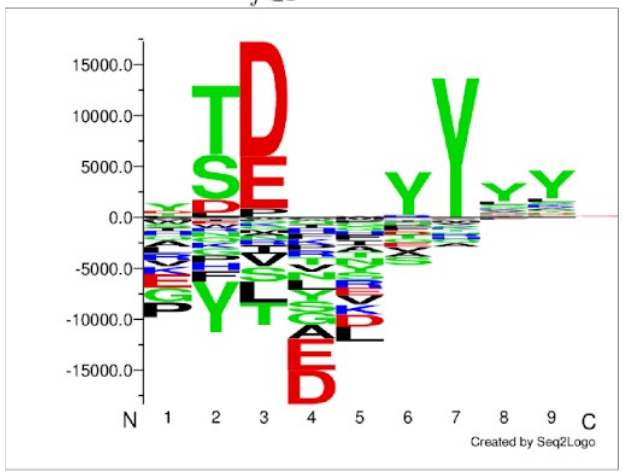




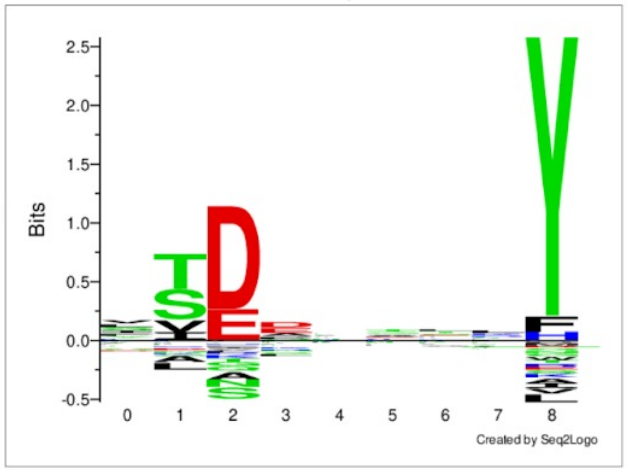
Bruno Alverez, manuscript in preparation

Filter (mis)alignment

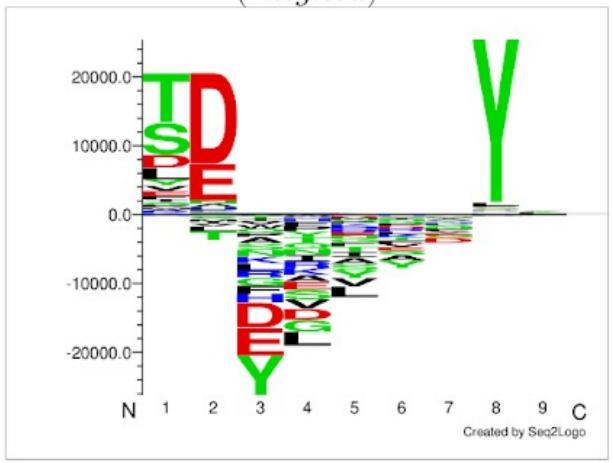
$\sum_{f \in F} \hat{\phi}_f$



NetMHCpan-4.1

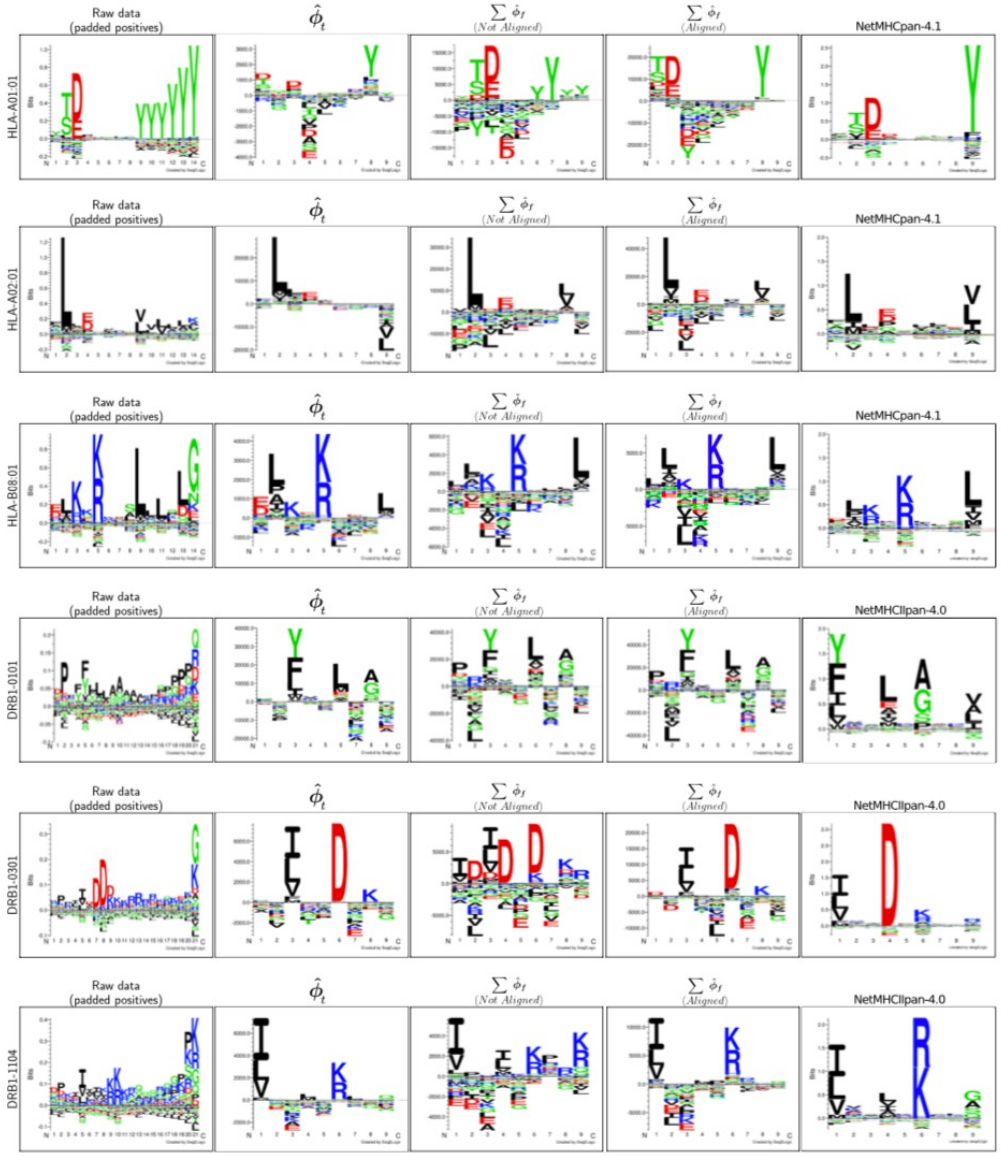


$\sum_{(Aligned)} \hat{\phi}_f$



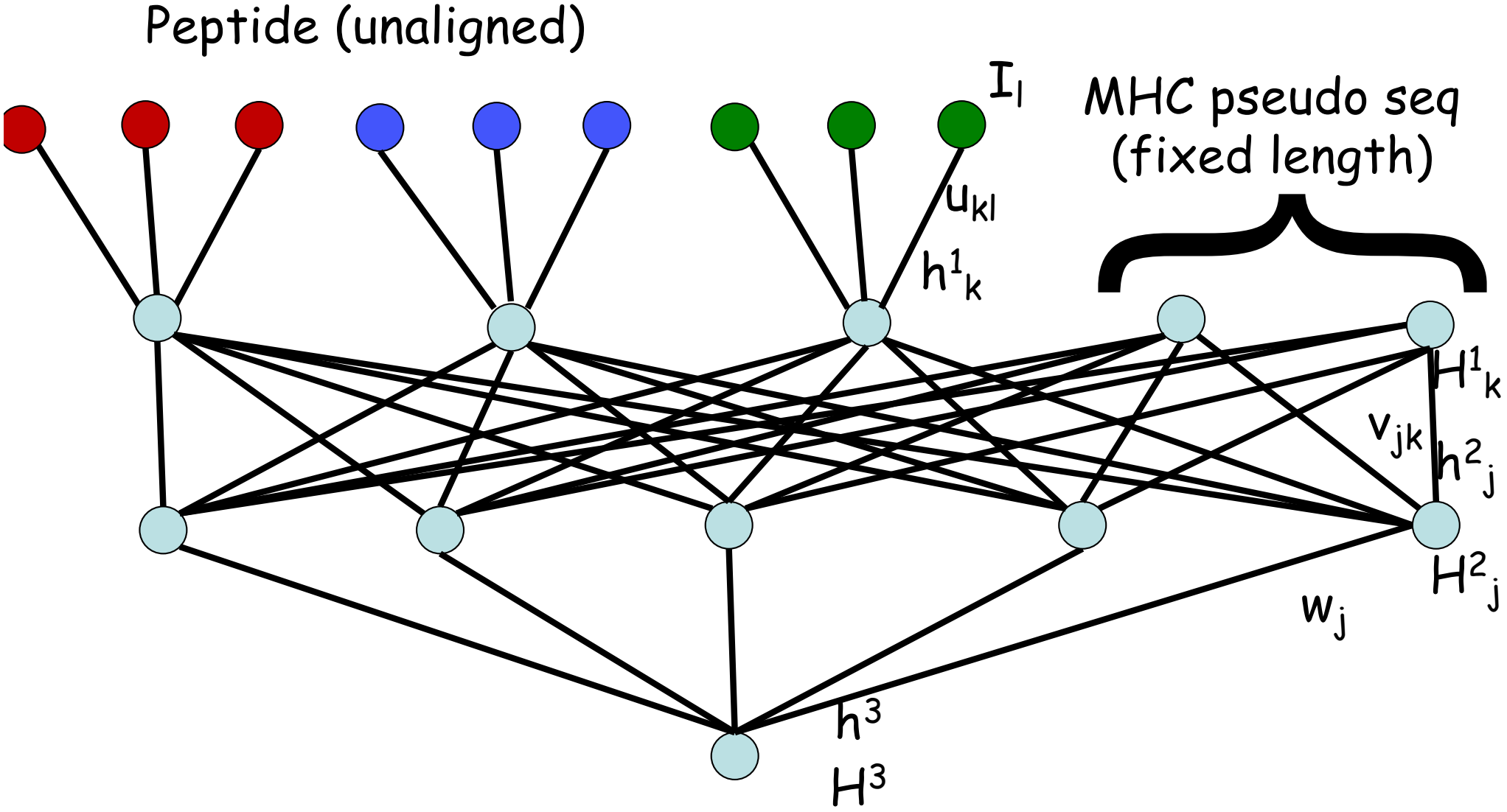
Bruno Alverez, manuscript in preparation

The full validation



Bruno Alverez, manuscript in preparation

Next step (add the essential hidden layer)



Bruno Alverez, manuscript in preparation

Modifying the CNN layer functionality

- Implement different max-pooling schemes depending on target label
- Include attention to capture conservation information and guide the selection of max-pool location
- Implementation of max-pooling restraints and/or filter initialization to capture prior binding motif knowledge
- ….

Thank you