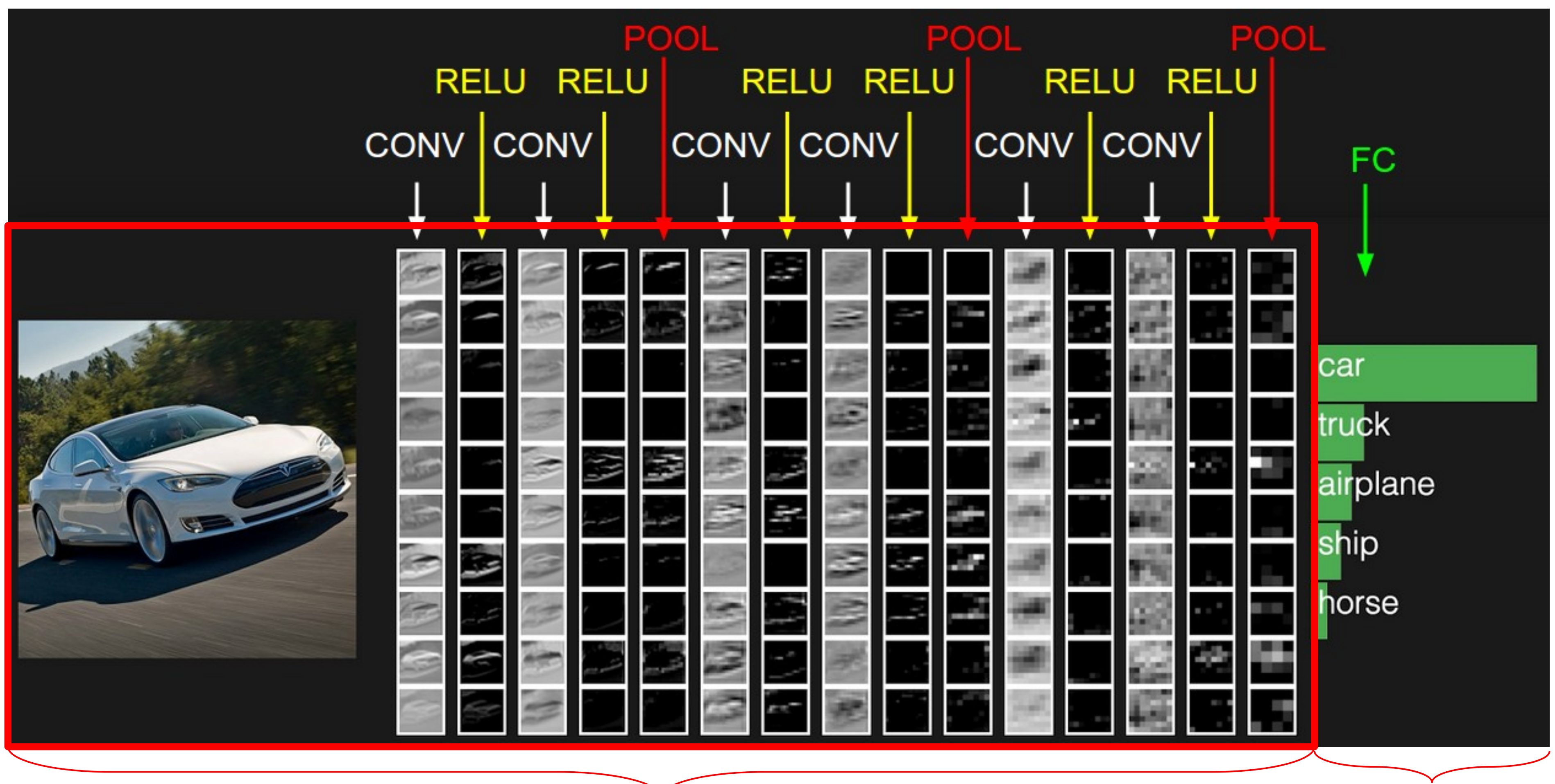
Neural Citation Network for Context-Aware Citation Recommendation

JIN WENHUI

Conv-Net Architecture

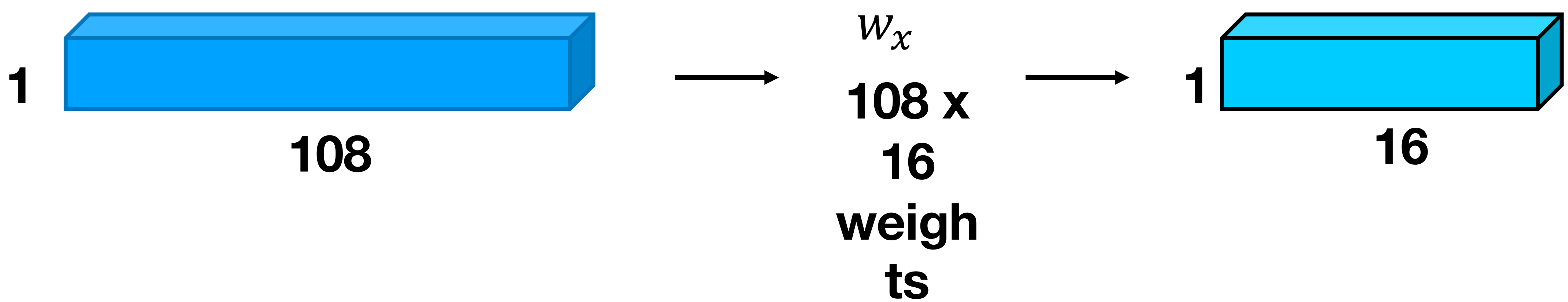


Conv-Net layers

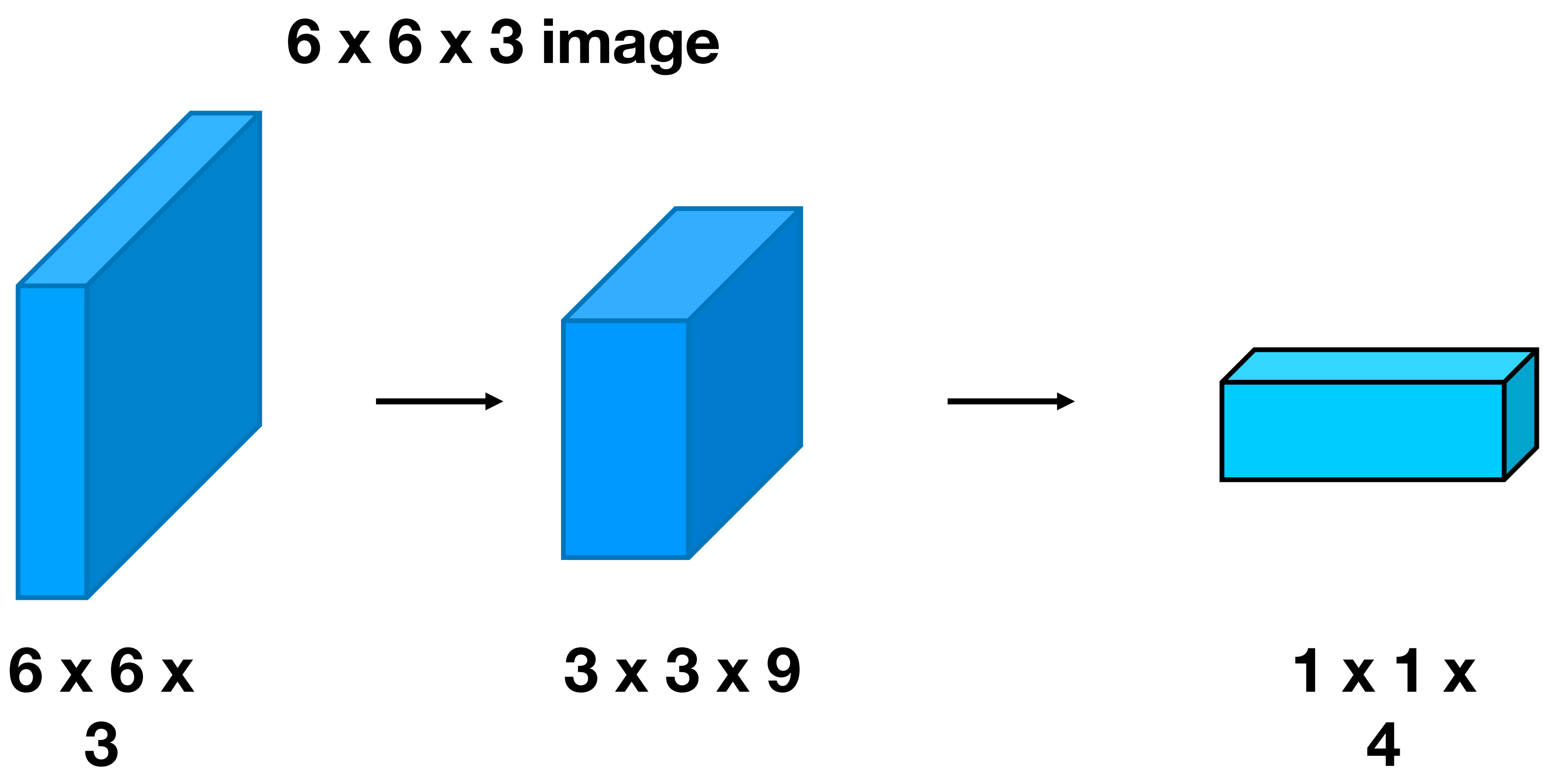
FC-Network

Fully Connected Layer

6 x 6 x 3 image -> stretch to 108 x 1



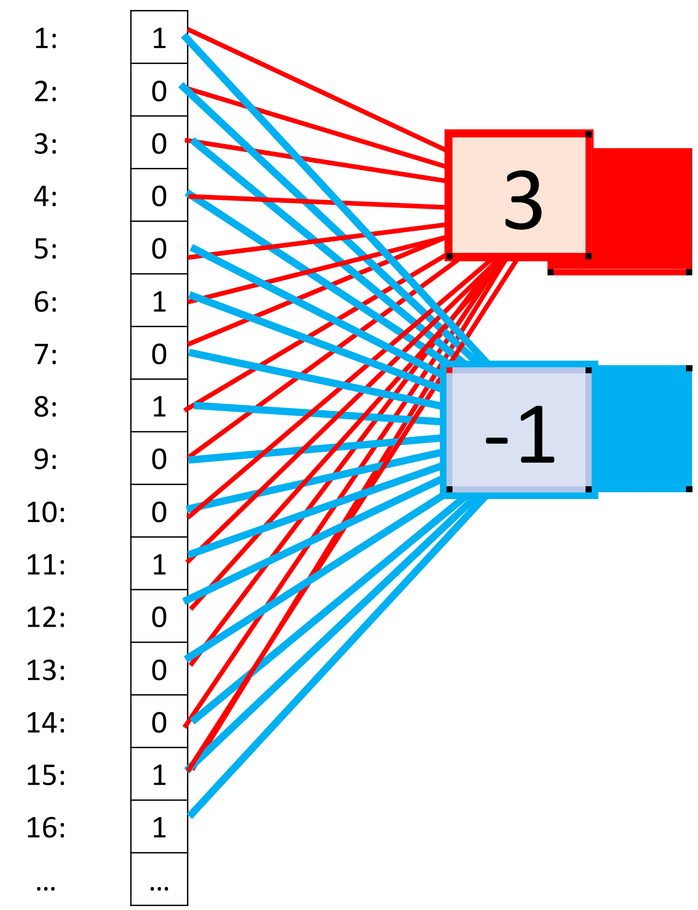
Convolutional Layer



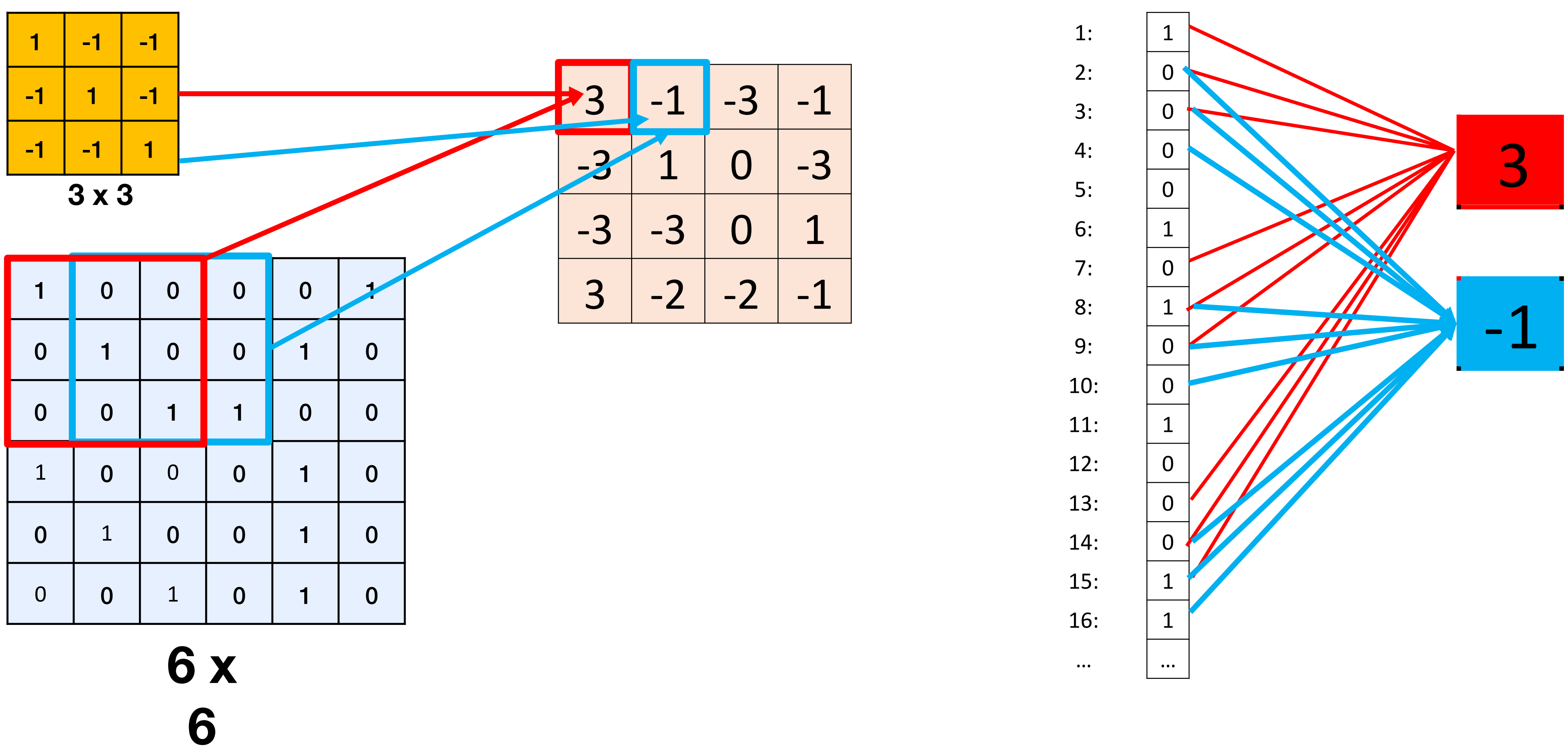
Why ConNet for image

1	0	0	0	0	1
0	1	0	0	1	0
0	0	1	1	0	0
1	0	0	0	1	0
0	1	0	0	1	0
0	0	1	0	1	0

6 x 6



Why ConNet for image



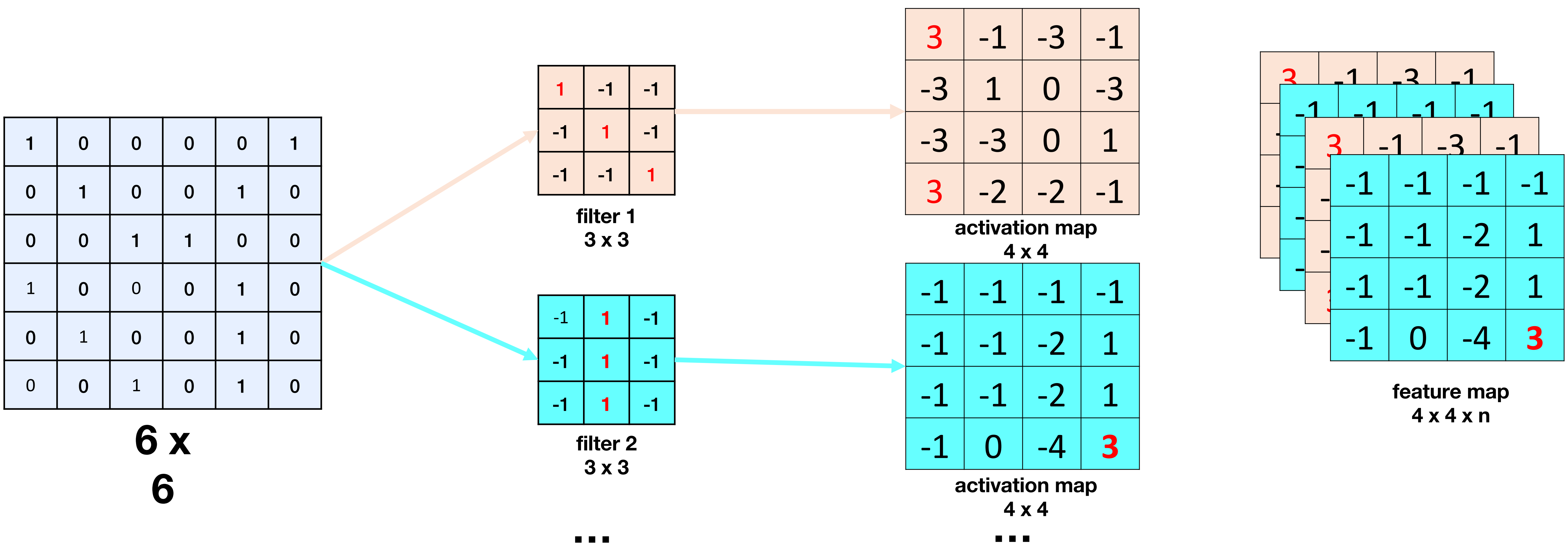
Input: 6 x 6 x 1

Filter: 3 x 3 x 1

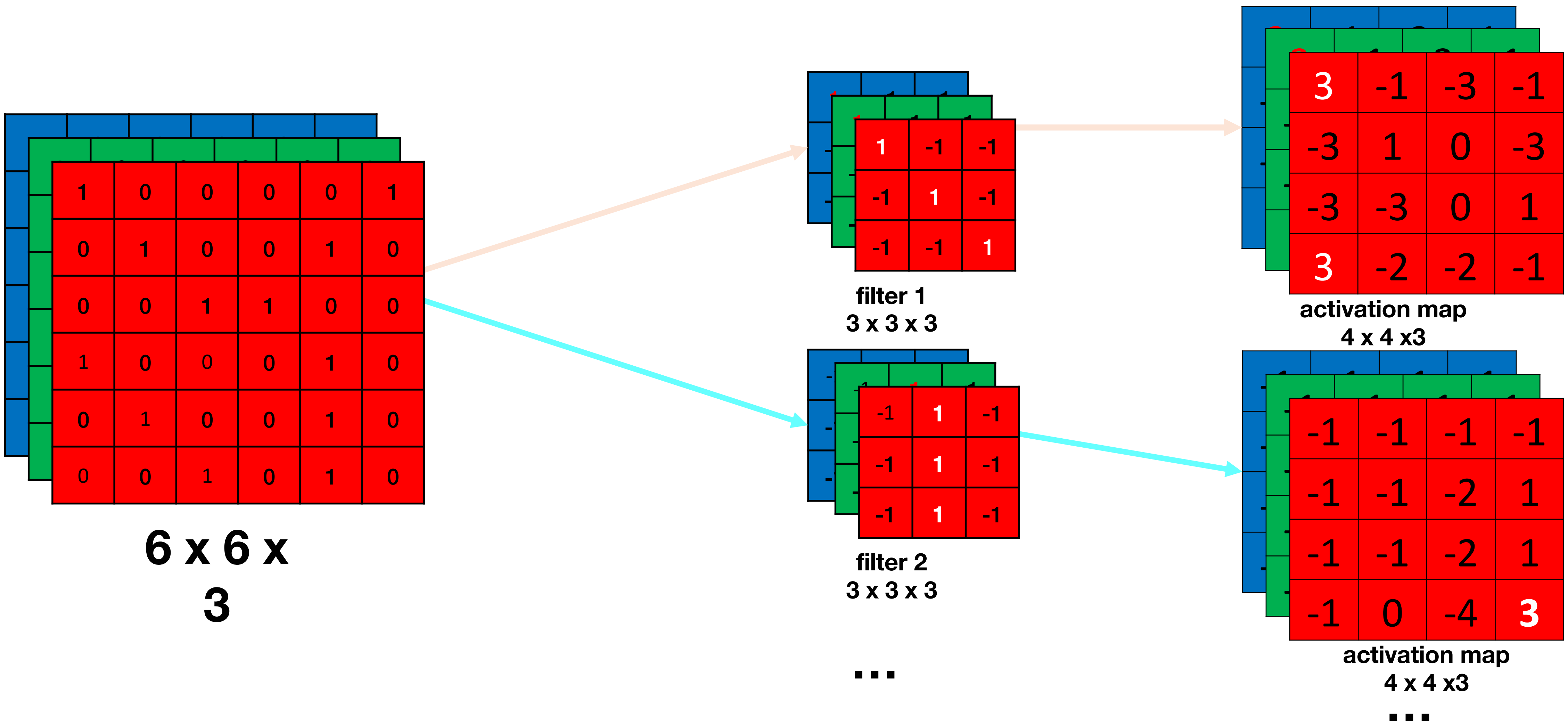
Stride : 1

Activation Map : 4 x 4 x 1

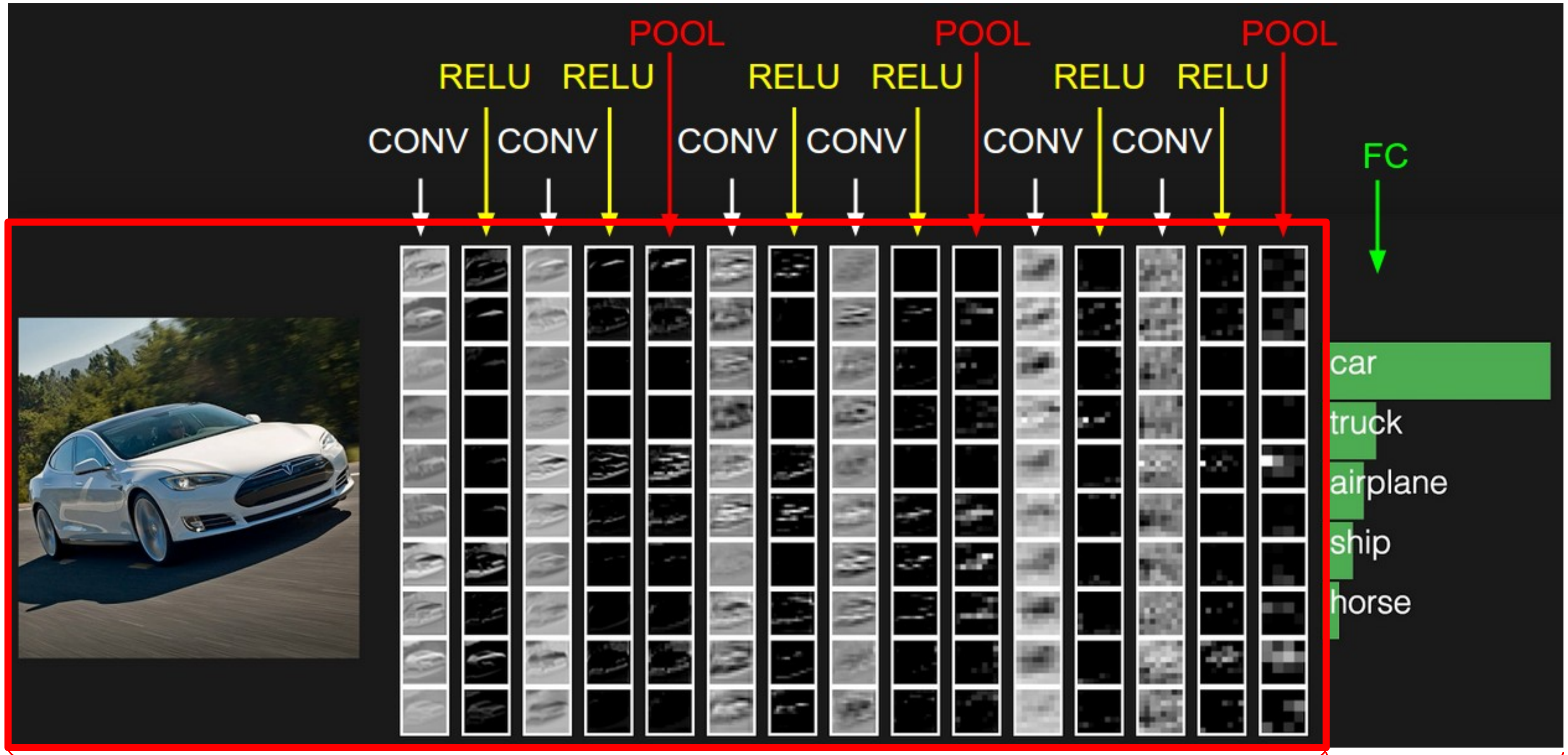
Why ConNet for image



Why ConNet for image



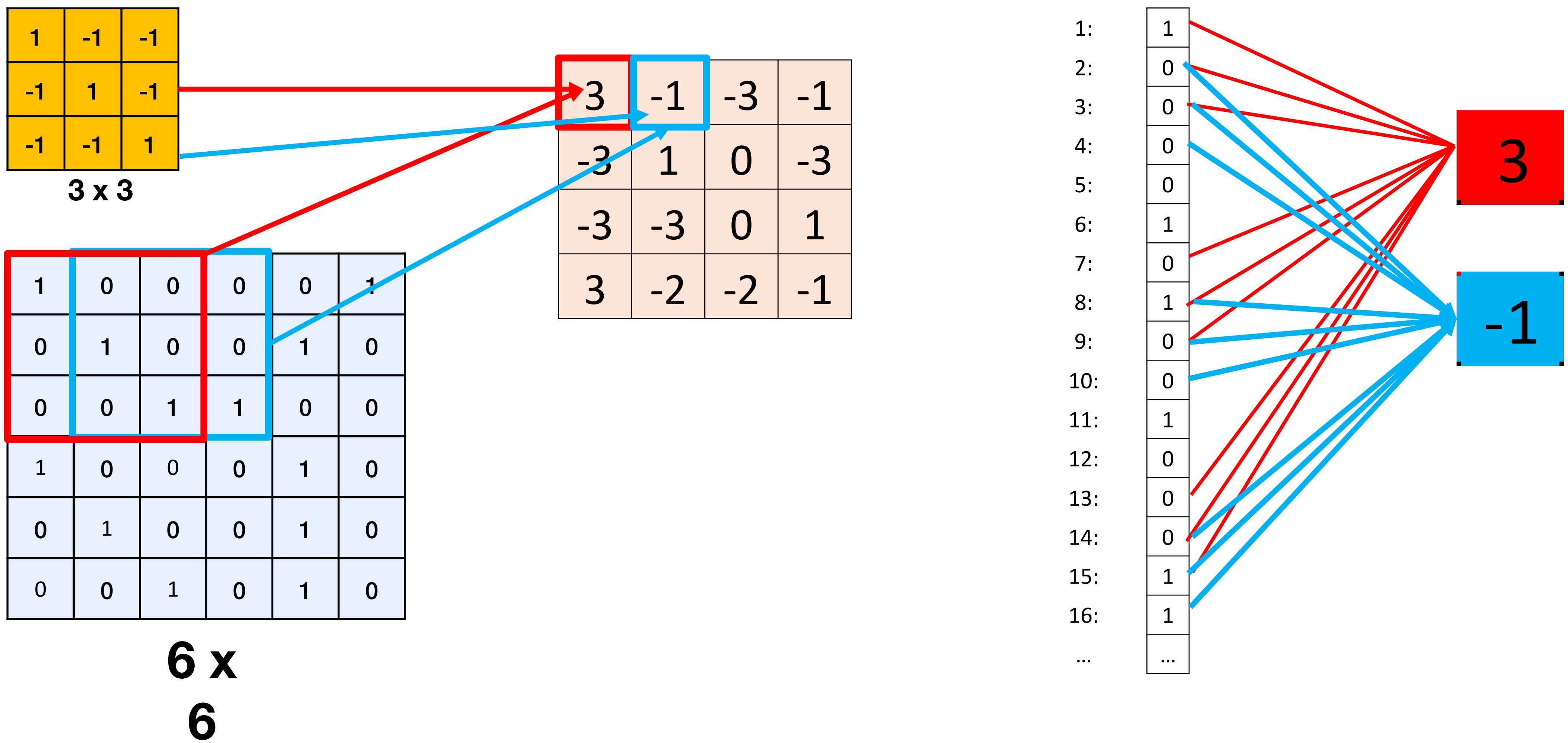
Conv-Net Architecture



Conv-Net layers

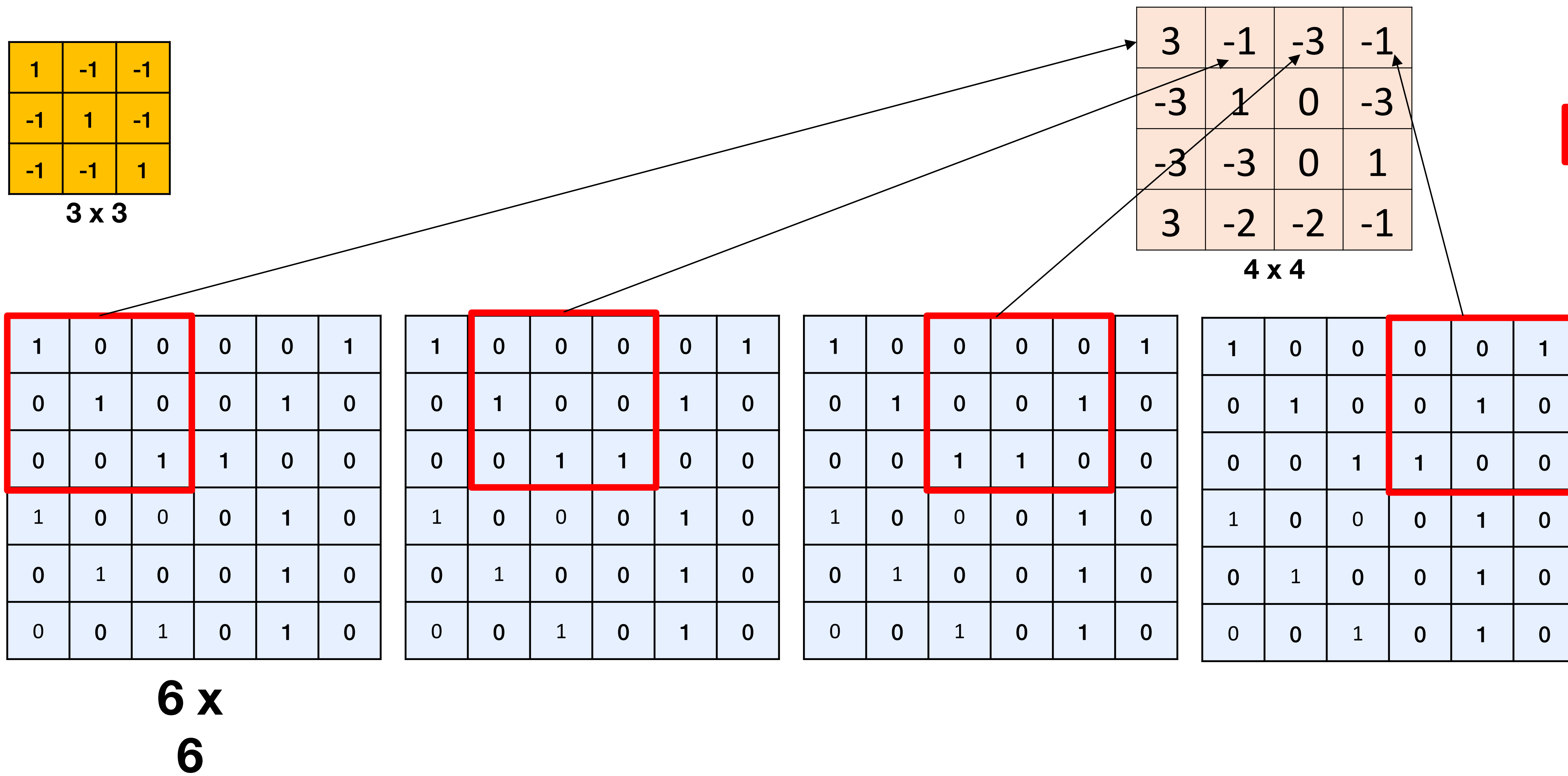
FC-Network

Spatial Arrangement



Input: 6 x 6 x 1
Filter: 3 x 3 x 1
Stride : 1
Activation Map : 4 x 4 x 1

Spatial Arrangement



Input: 6 x 6 x 1

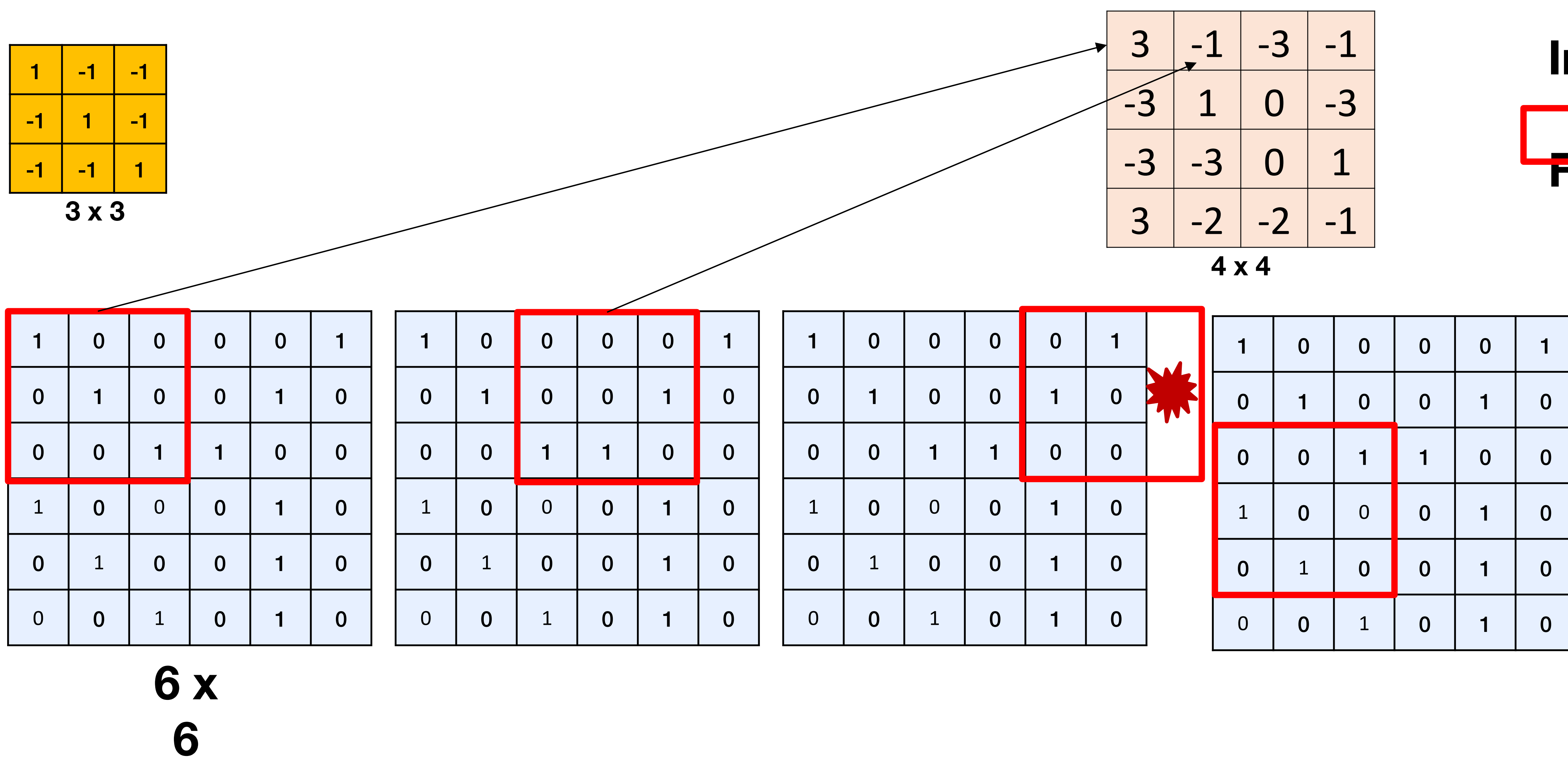
Filter: 3 x 3 x 1

Stride : 1

Activation Map : 4 x 4 x 1

...

Spatial Arrangement



Input: 6 x 6 x 1

Filter: 3 x 3 x 1

Stride : 2

Activation Map : ? x ? x 1

Output size

- Output size :
- Size = (N - F) / stride + 1

e.g. N = 6, F = 3;

stride = 1 => size = (6 - 3) / 1 + 1 = 3

stride = 2 => size = (6 – 3) / 2 + 1 = 2.5

e.g. N = 6, F = 2;

stride = 1 => size = (6 - 2) / 1 + 1 = 5

stride = 2 => size = (6 - 2) / 2 + 1 = 3

stride = 3 => size = (6 - 2) / 3 + 1 = 2.33..

1	0	0	0	0	1
0	1	0	0	1	0
0	0	1	1	0	0
1	0	0	0	1	0
0	1	0	0	1	0
0	0	1	0	1	0

Spatial Arrangement

1	-1	-1
-1	1	-1
-1	-1	1

3 x 3

1	0	1	0	1	0	1
1	0	0	0	0	1	0
0	1	0	0	1	0	0
0	0	1	1	0	0	1
0	0	1	0	1	0	1
0	1	0	0	1	0	0
1	0	0	0	1	0	0

7 x 7

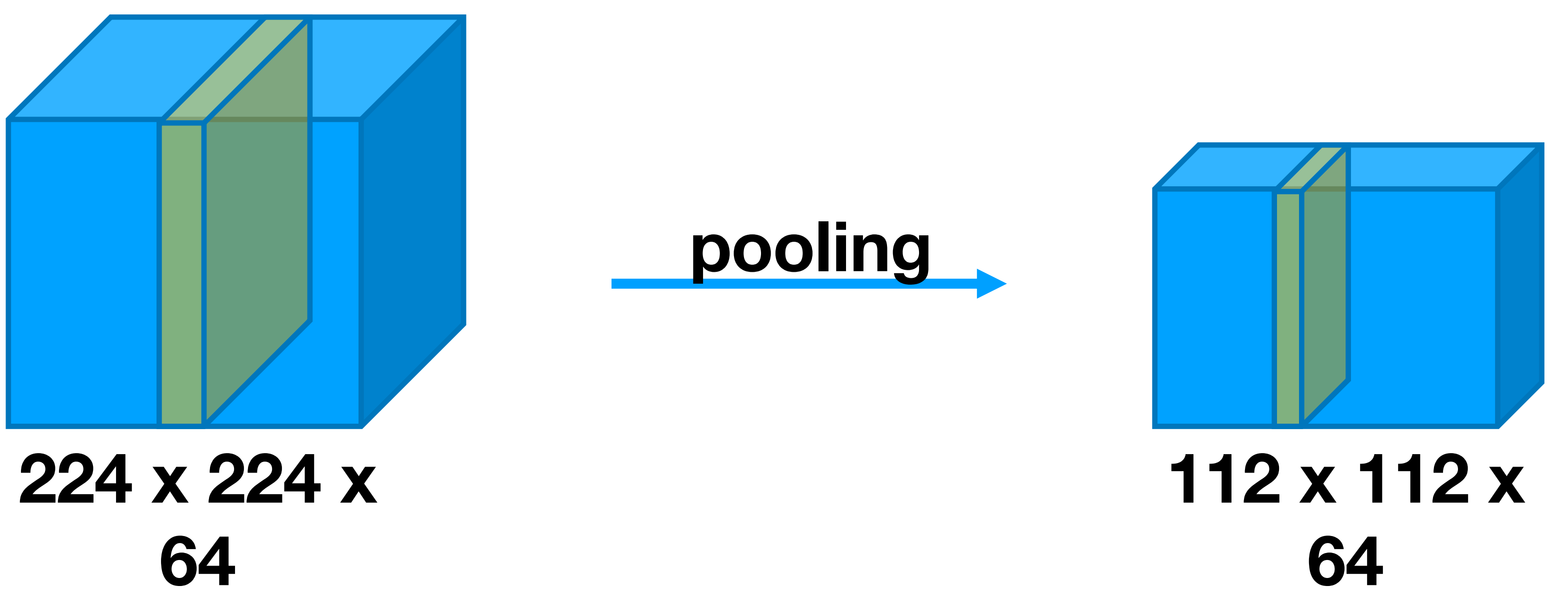
padding →

0	0	0	0	0	0	0	0	0
0	1	0	1	0	1	0	1	0
0	1	0	0	0	0	1	0	0
0	0	1	0	0	1	0	0	0
0	0	0	1	1	0	0	1	0
0	0	0	1	0	1	0	1	0
0	0	1	0	0	1	0	0	0
0	1	0	0	0	1	0	0	0
0	0	0	0	0	0	0	0	0

9 x 9

Input: 9 x 9 x 1
Filter: 3 x 3 x 1
Stride : 3
Activation Map : ? x ? x 1

Pooling layer



Max Pooling

1	-1	-1
-1	1	-1
-1	-1	1

filter 1

-1	1	-1
-1	1	-1
-1	1	-1

filter 2

3	-1	-3	-1
-3	1	0	-3
-3	-3	0	1
3	-2	-2	-1

-1	-1	-1	-1
-1	-1	-2	1
-1	-1	-2	1
-1	0	-4	3

Max Pooling

1	-1	-1
-1	1	-1
-1	-1	1

filter 1

-1	1	-1
-1	1	-1
-1	1	-1

filter 2

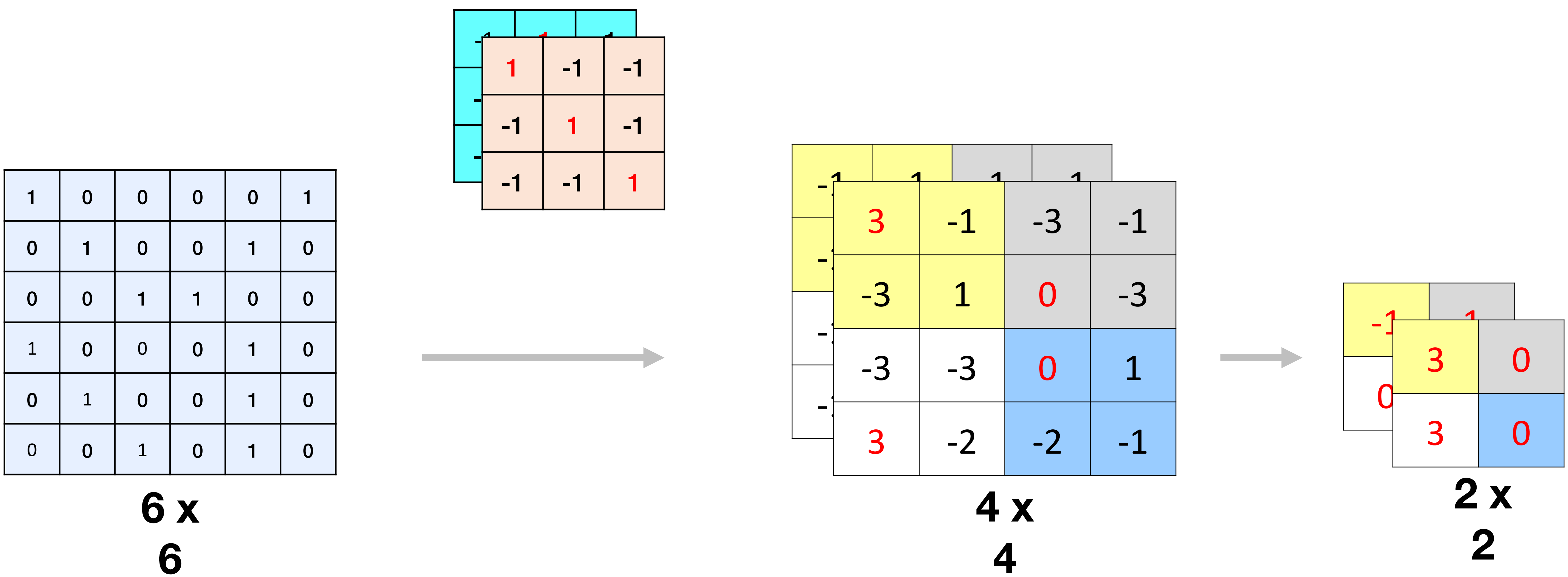
3	-1	-3	-1
-3	1	0	-3
-3	-3	0	1
3	-2	-2	-1

3	0
3	0

-1	-1	-1	-1
-1	-1	-2	1
-1	-1	-2	1
-1	0	-4	3

-1	1
0	3

Max Pooling



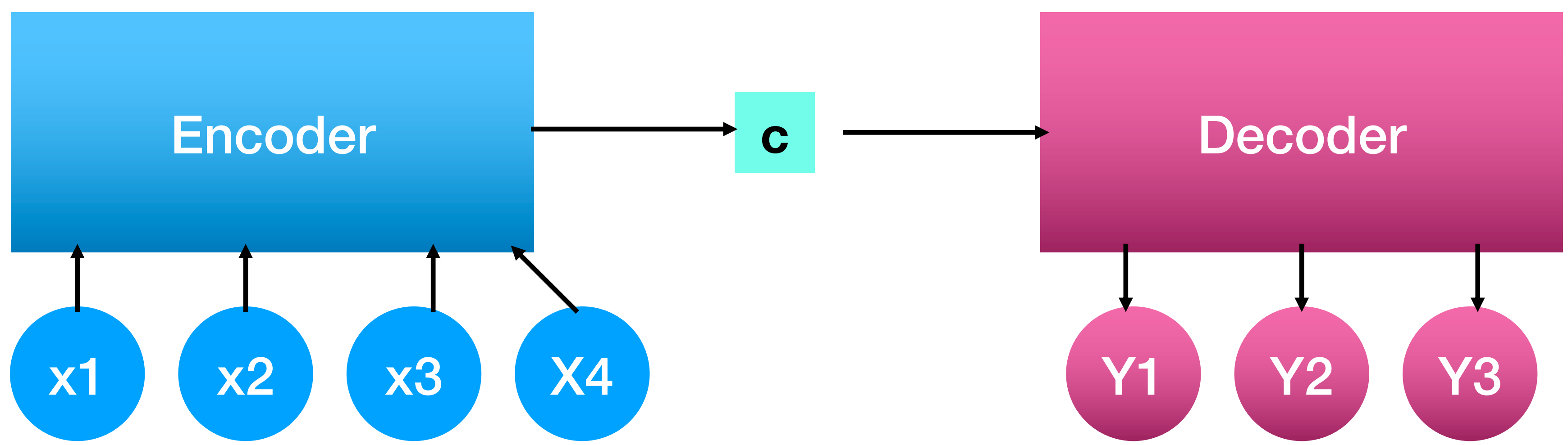
1 0 0 0 0 1
0 1 0 0 1 0
0 0 1 1 0 0
1 0 0 0 1 0
0 1 0 0 1 0
0 0 1 0 1 0
6 x 6
1 -1 -1
-1 1 -1
-1 -1 1
3 -1 -3 -1
-3 1 0 -3
-3 -3 0 1
3 -2 -2 -1
4 x 4
3 0
3 0
2 x 2

Review

CNN : we don`t need to calculate all the input data just want to compute a local regions input.

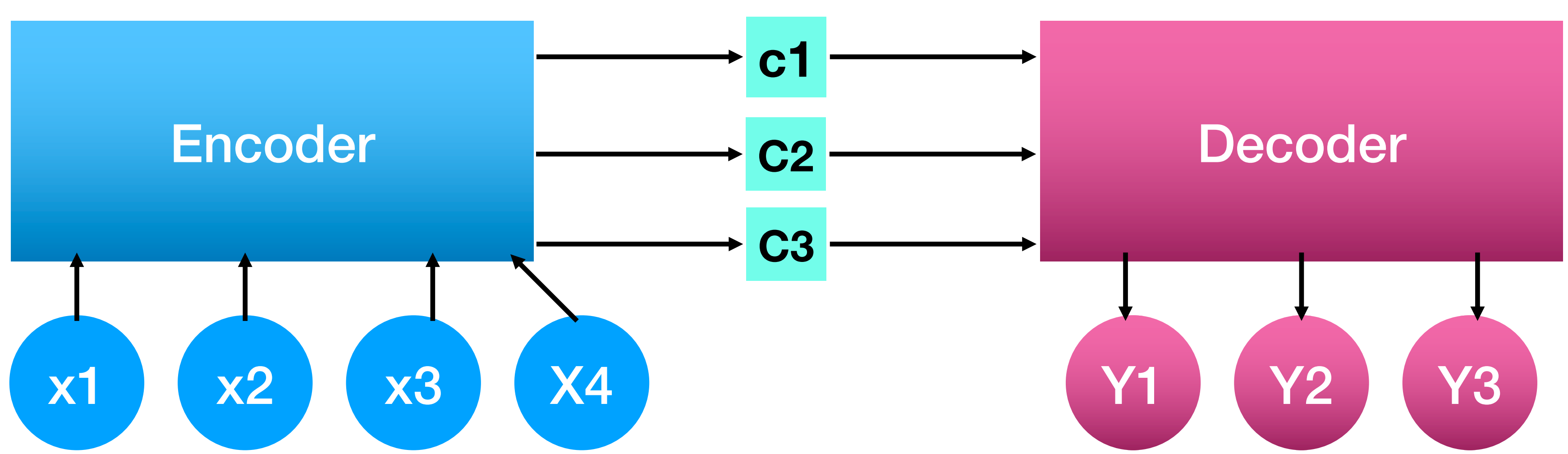
- Feature maps represent each features we extract from the input data.
- Max pooling : we select the most related features from the feature map.

Encoder-Decoder model :



Attention Model

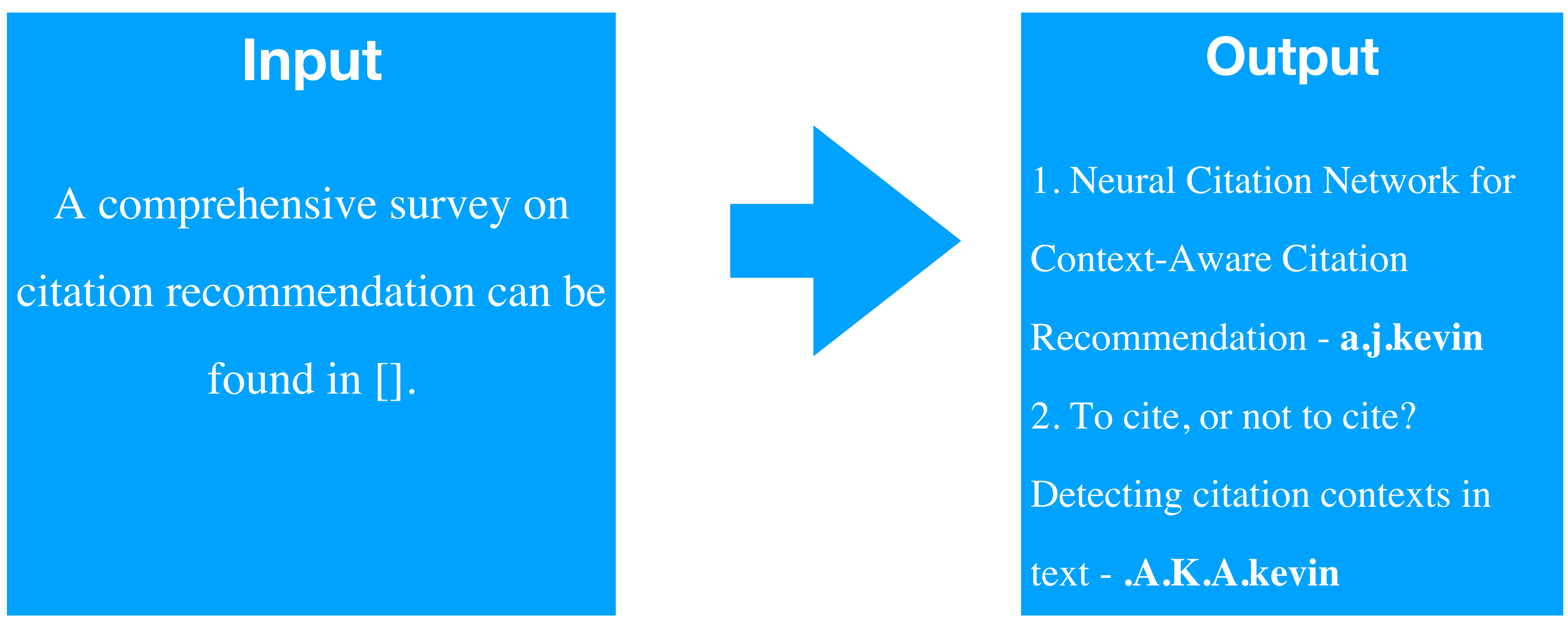
Attention parameter c1, c2, c3 tell us how much attention. How much the context depend on the features

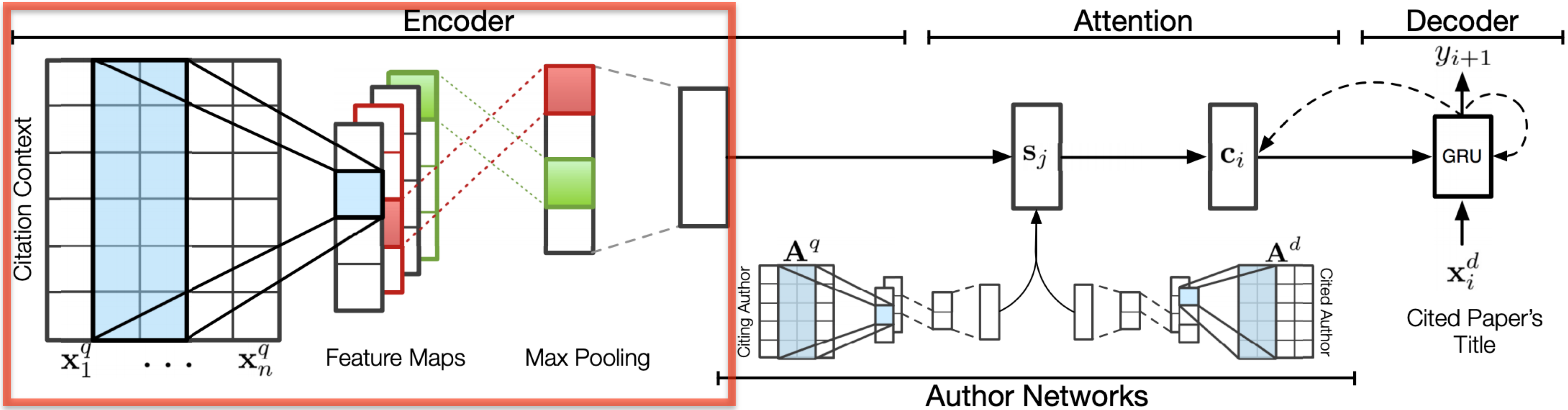


Abstract

Neural Citation Network(NCN) : increase the <u>**context-aware citation recommendation**</u> performance.

Current research based on bag-of-word representation that lack of valuable **semantics**.



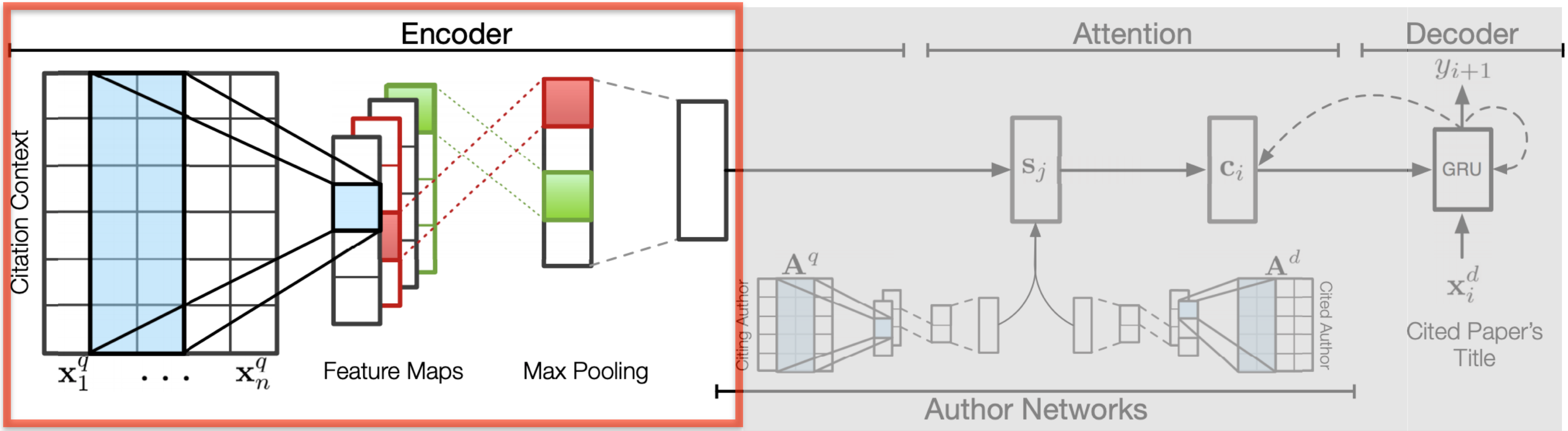


Citation Context → **Encoder** → **Attention** → **Decoder** → **Cited Paper`s Title**

Author network

Convolutional Neural Network

Recurrent Neural Network



Convolutional Neural Network

Input : n length context,
g dimensional word

Let x_t^q as a t-th word in citation context.

$\mathbf{x}_{1:n}^q = \mathbf{x}_1^q \oplus \ldots \oplus \mathbf{x}_n^q$ Denote concatenation of the embedding from 1 to n.

Active function : ReLU

Pooling : Max Pooling

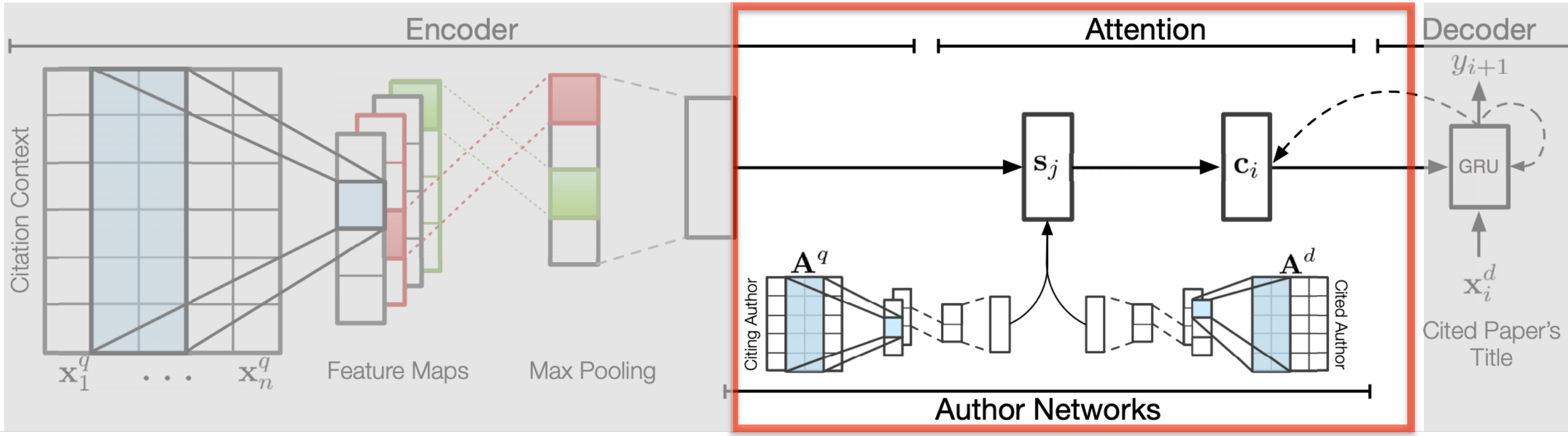
FIlter : l x g

$\{\mathbf{x}_{1:l}^q, \mathbf{x}_{2:l+1}^q, \ldots, \mathbf{x}_{n-l+1:n}^q\}$

After extract features from Max Pooling layer
Send to FC layer to make these features interaction.

$$\mathbf{s}_j = \tanh(\mathbf{U}_{s_j} \hat{\mathbf{o}}_j + \mathbf{b}_{s_j})$$

PS: TDNN each data is separately and here allowing all feature maps to be computed in parallel

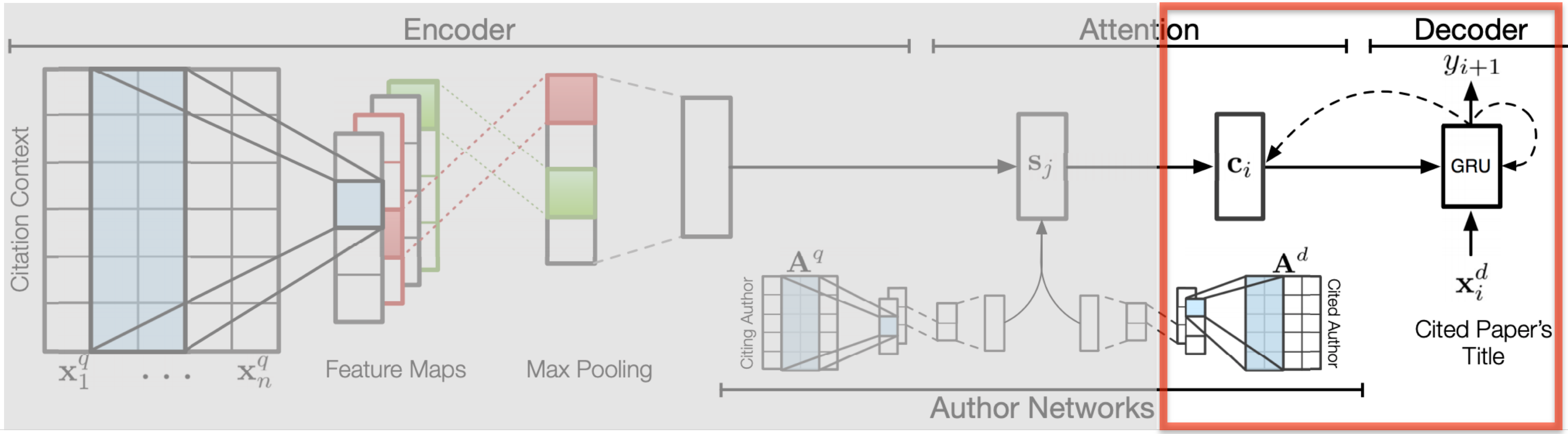


NCN is capable of characterizing the semantic composition of <u>citation contexts</u> and corresponding <u>cited documents title</u> by exploiting ***<u>author relations</u>***.

To capture the most prominent author, here consider both the citing(context) and cited(title) manuscript authors with a shared embedding space, but learn two separate TDNN.

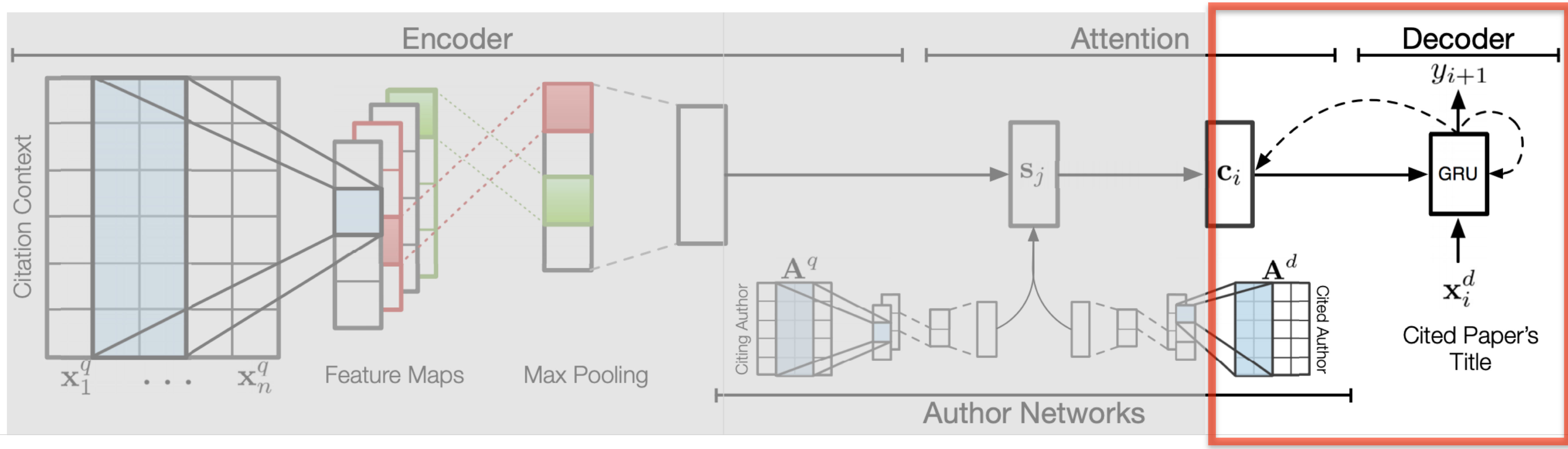
So, in here Attention model, it not only focus on the features from the citation context but also consider the author`s representation.

$$\mathbf{s}_j = [f(\mathbf{X}^q) \oplus f(\mathbf{A}^q) \oplus f(\mathbf{A}^d)]_j$$



Although the max pooling layer obtains the most relevant features but max pooling just select the most related word and drop surrounding words.
So Attention mechanism learns a weighted interpolation c_i dependent on all of the encoder`s representation and also need the decoder state to obtaining a richer representation : where a_{ij} is the alignment between the i-th word and the j-th output

$$\mathbf{c}_i = \sum_j \alpha_{ij}\mathbf{s}_j \qquad \text{where } \alpha_{ij} = \text{softmax}(\mathbf{v}^{\mathsf{T}} \tanh(\mathbf{W}_a \mathbf{h}_{i-1} + \mathbf{U}_a \mathbf{s}_j))$$



CNN precess every data as a separately object Then RNN think they have a sequence and relations.

x_i^d is a e-dimensional embedding corresponding to the i-th word of the cited document`s title.

$\mathbf{W}_{[z,r,o]}, \mathbf{V}_{[z,r,o]}, \mathbf{U}_{[z,r,o]}$ are weight matrices to be learned,

$$\mathbf{z}_i = \sigma(\mathbf{W}_z \mathbf{x}_i^d + \mathbf{V}_z \mathbf{c}_i + \mathbf{U}_z \mathbf{h}_{i-1})$$
$$\mathbf{r}_i = \sigma(\mathbf{W}_r \mathbf{x}_i^d + \mathbf{V}_r \mathbf{c}_i + \mathbf{U}_r \mathbf{h}_{i-1})$$
$$\tilde{\mathbf{h}}_i = \tanh(\mathbf{W}_o \mathbf{x}_i^d + \mathbf{V}_o \mathbf{c}_i + \mathbf{r}_i \circ \mathbf{U}_o \mathbf{h}_{i-1})$$
$$\mathbf{h}_i = (1 - \mathbf{z}_i) \circ \tilde{\mathbf{h}}_i + \mathbf{z}_i \circ \mathbf{h}_{i-1}$$

Final result of RNN decoder :

$$P(y_i | y_{\le i}, \mathbf{s}) = \text{softmax}(\mathbf{V}\mathbf{h}_i)$$

$$\log P(\mathbf{y} | \mathbf{X}^{\mathbf{q}}, \mathbf{X}^{\mathbf{d}}, \mathbf{A}^{\mathbf{q}}, \mathbf{A}^{\mathbf{d}}) = \sum_i^m \log P(y_i | y_{\le i}, \mathbf{s})$$

EXPERIMENTS

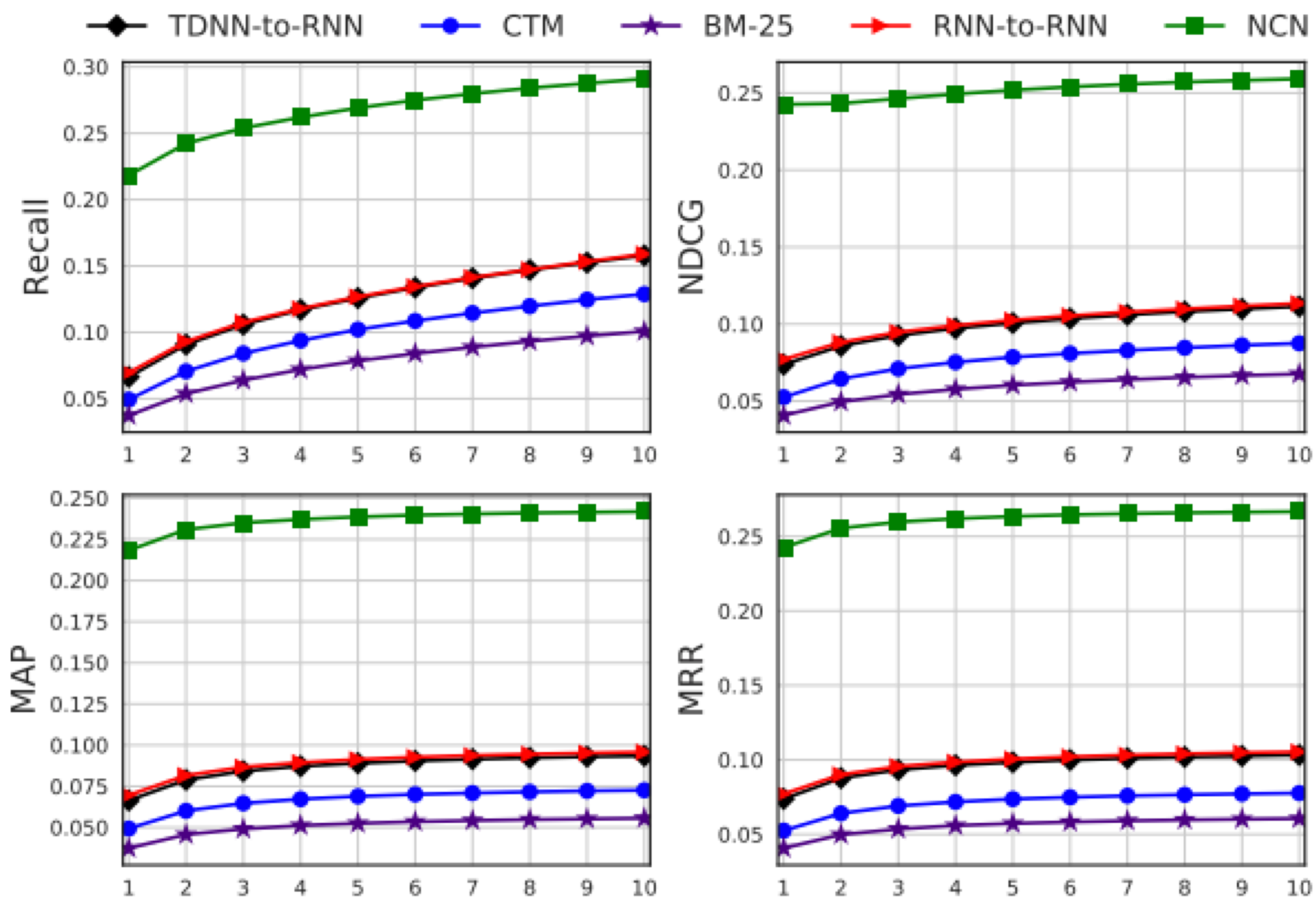


Figure 2: Recall, NDCG, MAP, and MRR as the number of recommendations vary from 1 to 10.

EXPERIMENTS

Context: "find a distribution over the latent variables that is close to the posterior of interest. Variational methods provide effective approximations in topic models and nonparametric Bayesian models"
Neural Citation Network
1. **Graphical models, exponential families, and variational inference** 2. Graphical models and variational methods 3. **An introduction to variational methods for graphical models**
CTM
1. Indexing by latent semantic analysis 2. **An introduction to variational methods for graphical models** 3. Bayesian data analysis
RNN-to-RNN
1. **An introduction to variational methods for graphical models** 2. The variational formulation of the Fokker–Planck equation 3. A Bayesian analysis of the multinomial probit model with fully identified parameters

Table 2: Top 3 recommendations for NCN, CTM and RNN-to-RNN for the citation context (query), correct recommendations are in bold.

Conclusion

- This paper propose a context-aware citation recommend system that based on neural citation network which also the author designed.
- The architect model is Encoding-Decoding model.
- To trade off the computation time and semantic diversity, in this paper, the author select the CNN variant TDNN as the Encoding method.
- And because the sequence property of words the decode part the author use the RNN model.
- Beside these, to better control the features in here use the attention model and use the author networks to get the better semantic.

Thank you ~