# Natural Language Processing

**Lecture 19**

Dirk Hovy

dirk.hovy@unibocconi.it

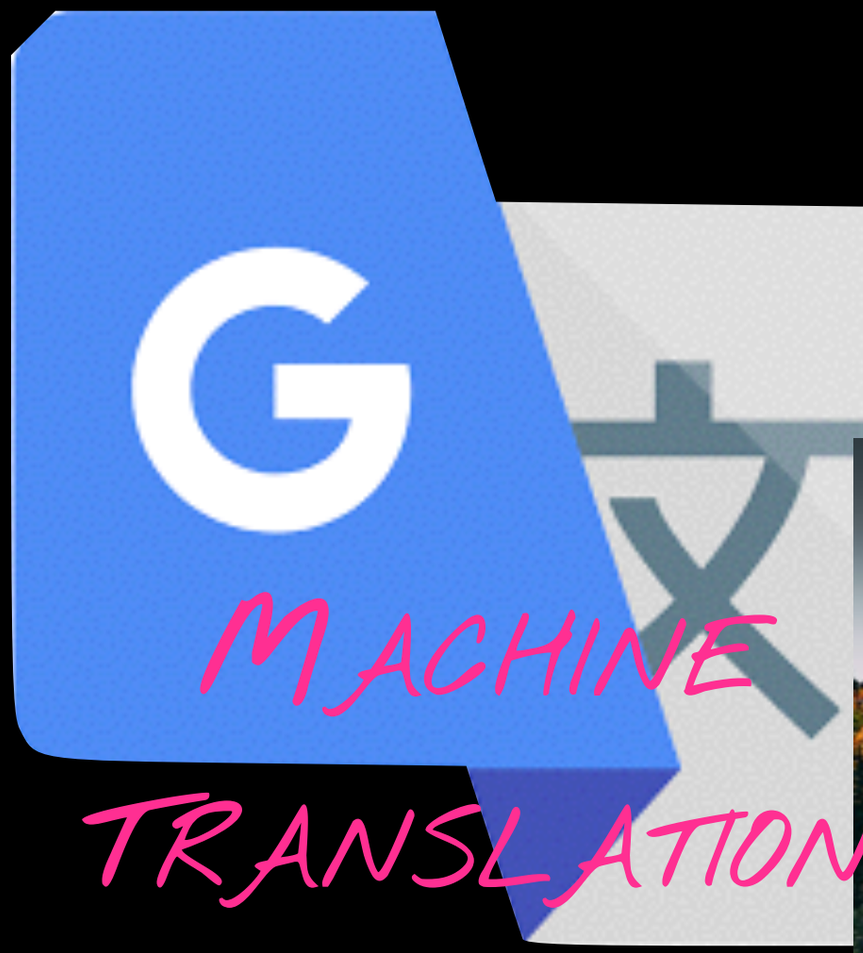
@dirk_hovy

Bocconi

# Goals for Today

- Learn about **recurrent neural network architectures**
- Learn about **convolutional neural network architectures**
- Understand the concept of **convolution** and **pooling**
- Understand the **difference between recurrent and convolutional** networks
- Understand the **attention mechanism**

# Recurring Matters



MACHINE TRANSLATION



IMAGE CAPTIONS



Bocconi

# Long-Term Trouble

SUBJECT

"Wenn er aber auf der Strasse der in Sammt und Seide gehüllten jetzt sehr ungenirt nach der neusten Mode gekleideten Regierungsräthin begegnet."

VERB

Mark Twain, *The Awful German Language*

# Long-Term Trouble

NEGATION

This is not in any sense of the word a funny movie.

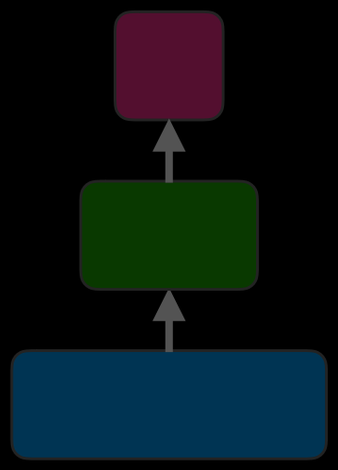
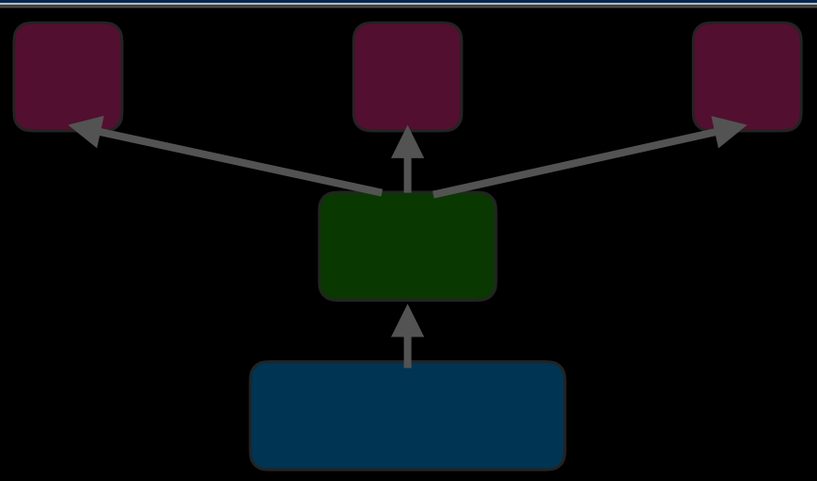
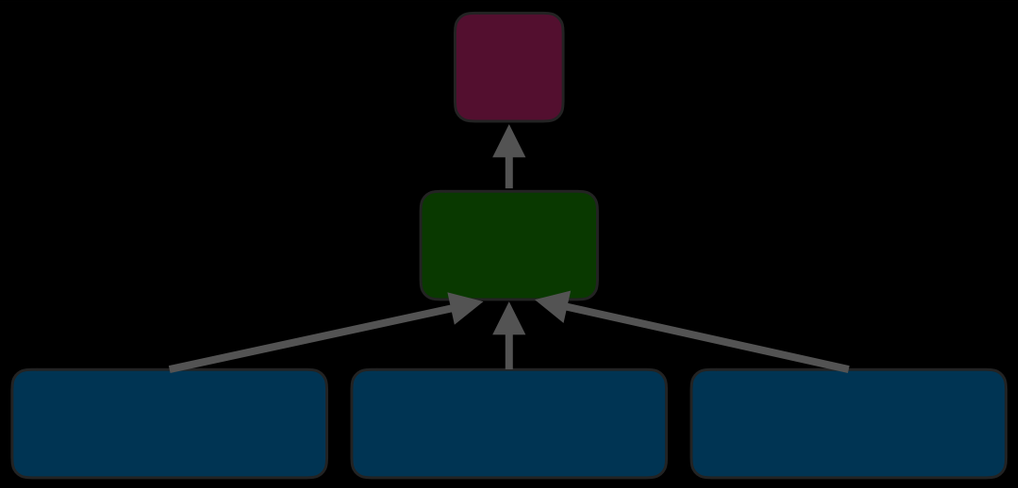
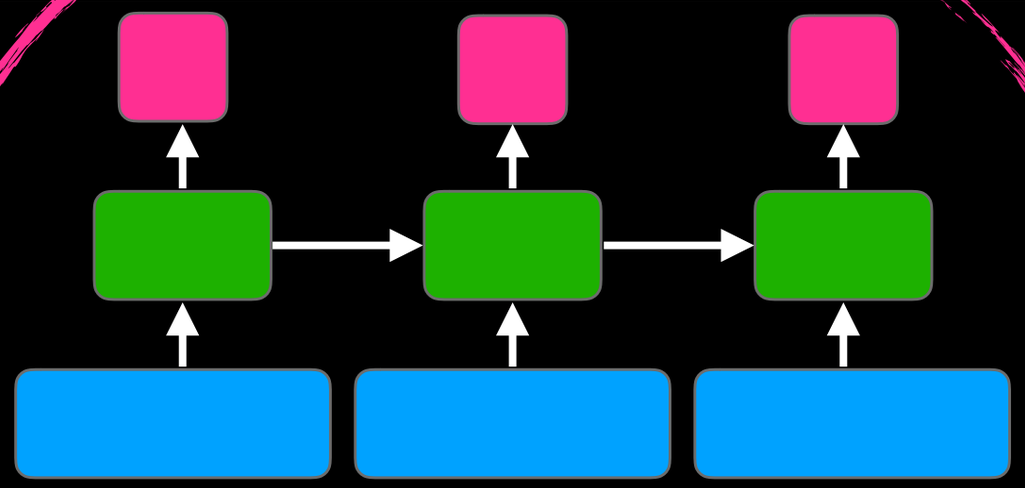
# Sequence Tagging

| **PRON** | **VERB** | **ADP** | **DET** | **???** | **PUNCT** |
|---|---|---|---|---|---|
| I | went | to | the | show | . |

**show** {VERB, NOUN}


Structured prediction: depends on the POS of a previous word

# Types of Text Classification

| | Fixed length | Variable length |
|---|---|---|
| **Fixed length** |  Logistic Regression, Perceptron, Feed-Forward Network, Random Forest, Naive Bayes, SVM, … |  Multitask Learning, Decoder |
| **Variable length** |  Convolutional Neural Networks (CNN) |  Recurrent Neural Networks (RNN), Hidden Markov Models (HMM), Conditional Random Fields |

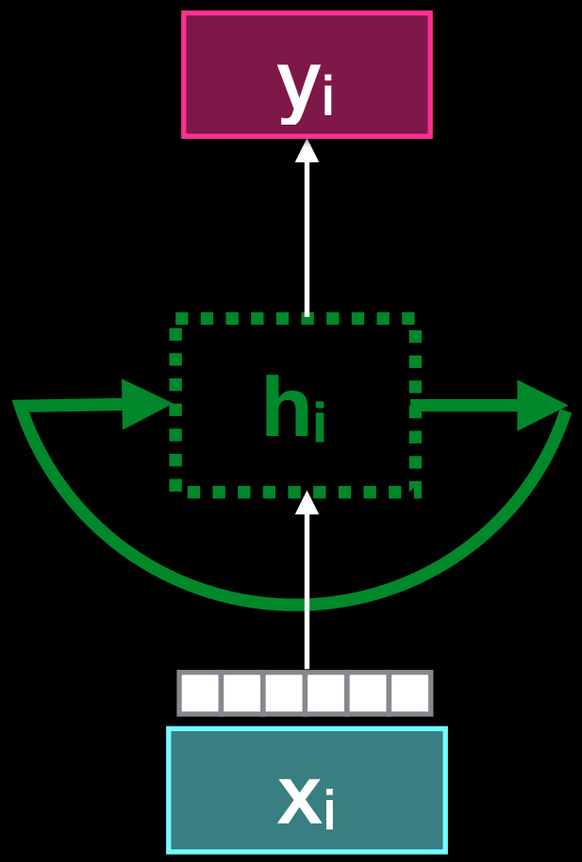
# Recurrent Networks

Bocconi

# Recurrence

$$y_i = f(h_i)$$

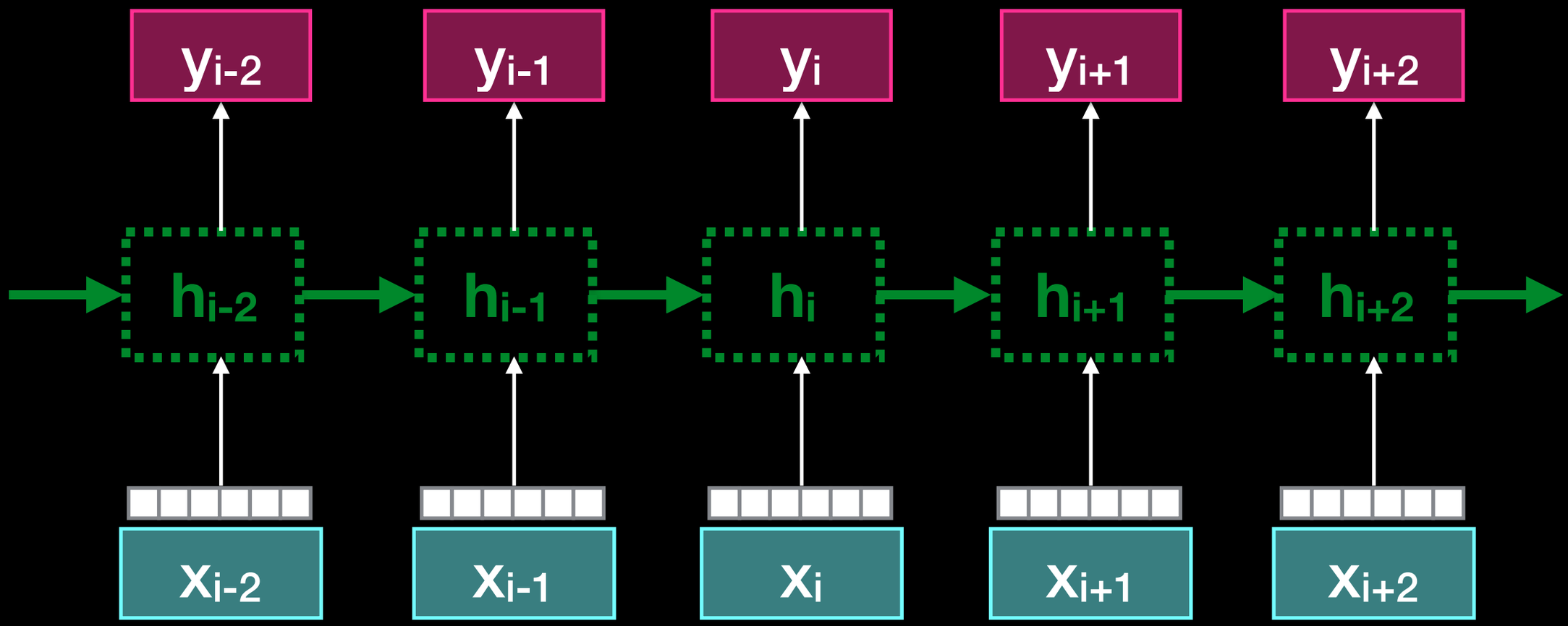
$$h_i = s(h_{i-1}, x_i)$$



# …Unrolled

$$y_i = f(h_i)$$

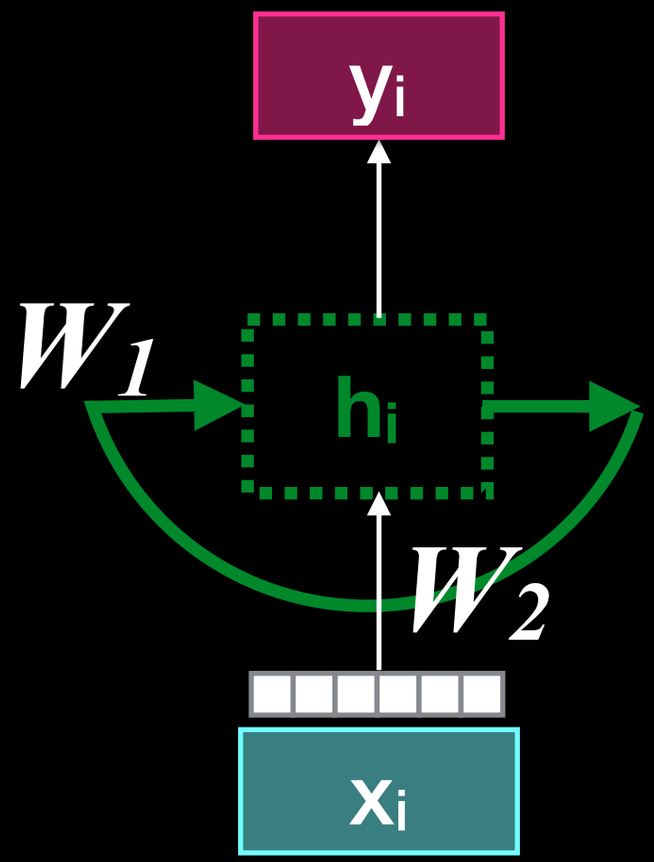
$$h_i = s(h_{i-1}, x_i)$$



# Concretely

$$y_i = f(h_i) = h_i$$

$$h_i = s(h_{i-1}, x_i) = tanh(W_1 h_{i-1} + W_2 x_i + b)$$




SIMPLE RNN

# Recap: LMs

TRIGRAM MODEL

$$P(w_1, w_2, \ldots, w_n) \approx \prod_{i=1}^{N} P(w_i | w_{i-2}, w_{i-1})$$

`* * The weather today is fine STOP`

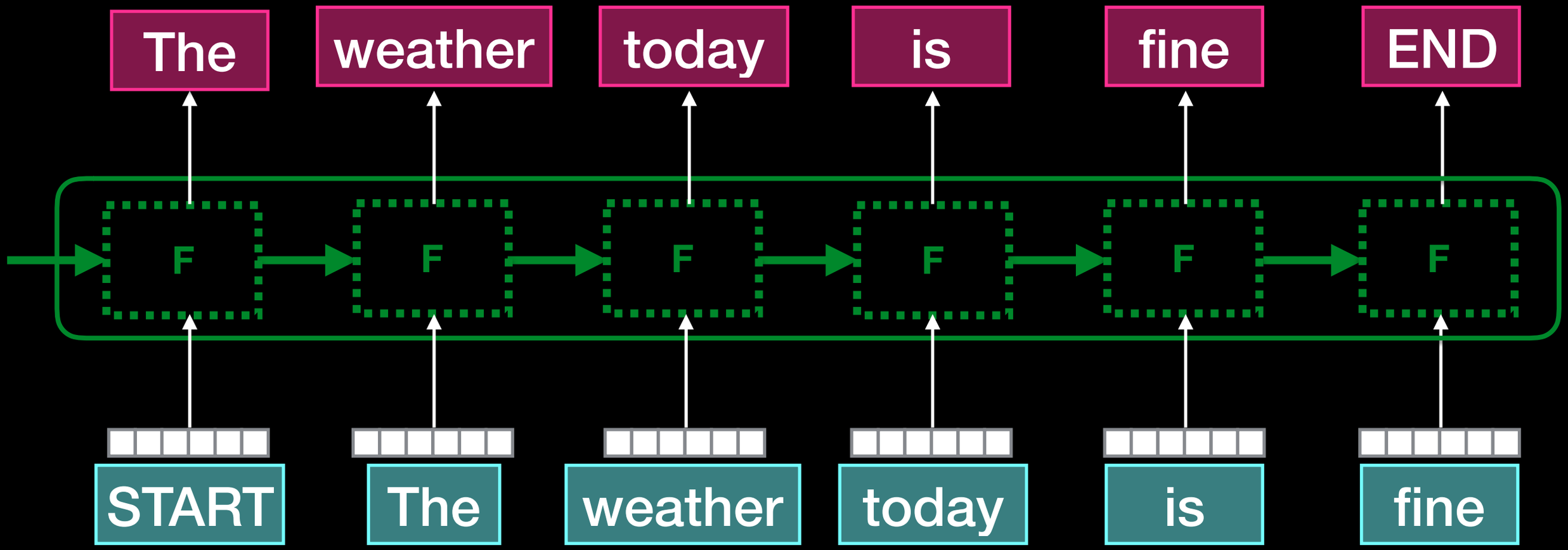
CHAIN RULE

$$P(S) = P(w_1, \ldots, w_n) = P(\texttt{The} \mid \texttt{* *})$$
$$\times P(\texttt{weather} \mid \texttt{* The})$$
$$\times P(\texttt{today} \mid \texttt{The weather})$$
$$\times P(\texttt{is} \mid \texttt{weather today})$$
$$\times P(\texttt{fine} \mid \texttt{today is})$$
$$\times P(\texttt{STOP} \mid \texttt{is fine})$$

# Neural LMs

$$P(w_1, w_2, \ldots, w_n) \approx \prod_{i=1}^{N} P(w_i | w_1 \ldots w_{i-1})$$

INIFINTE MODEL



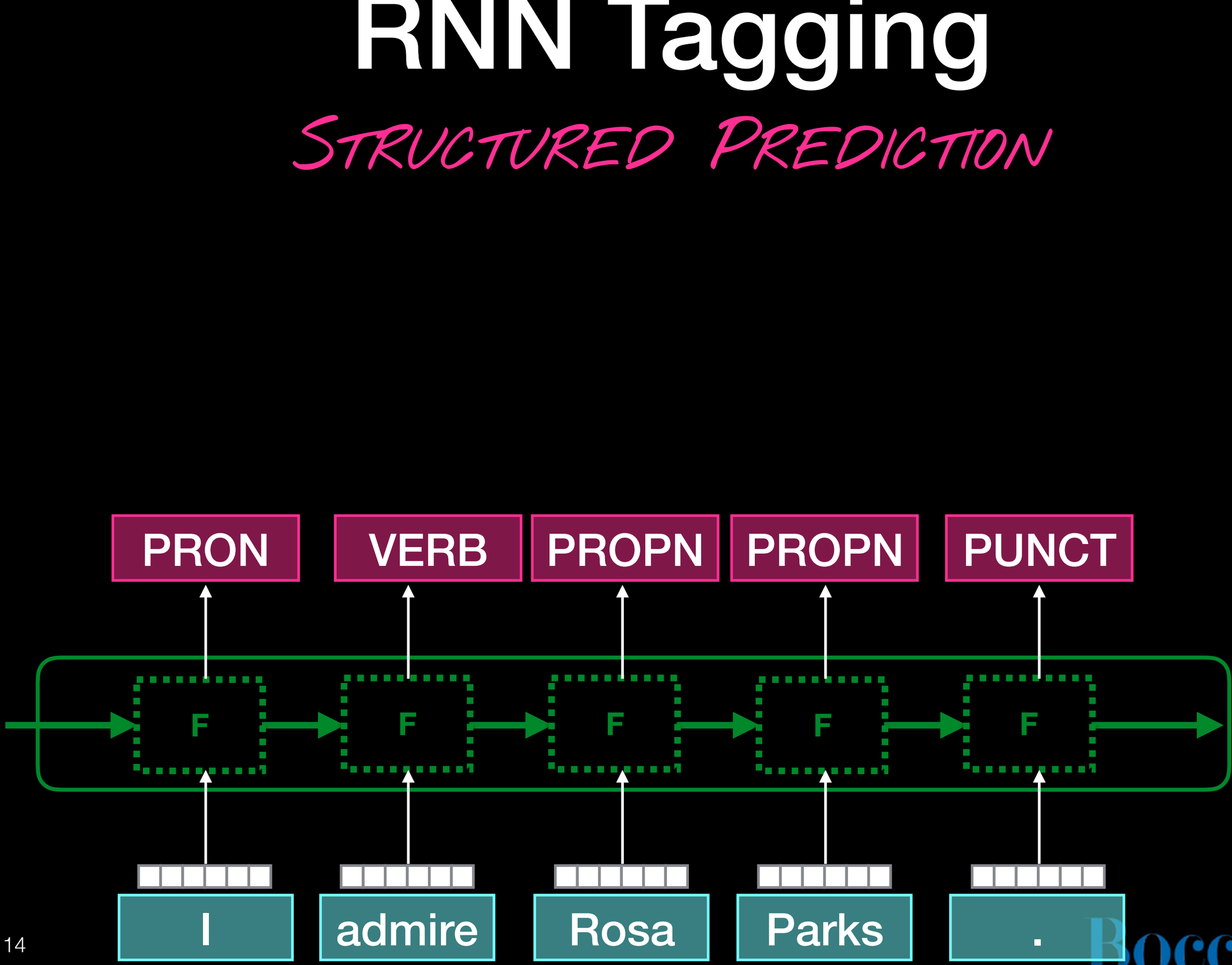

PREDICT NEXT WORD GIVEN HISTORY



Bocconi

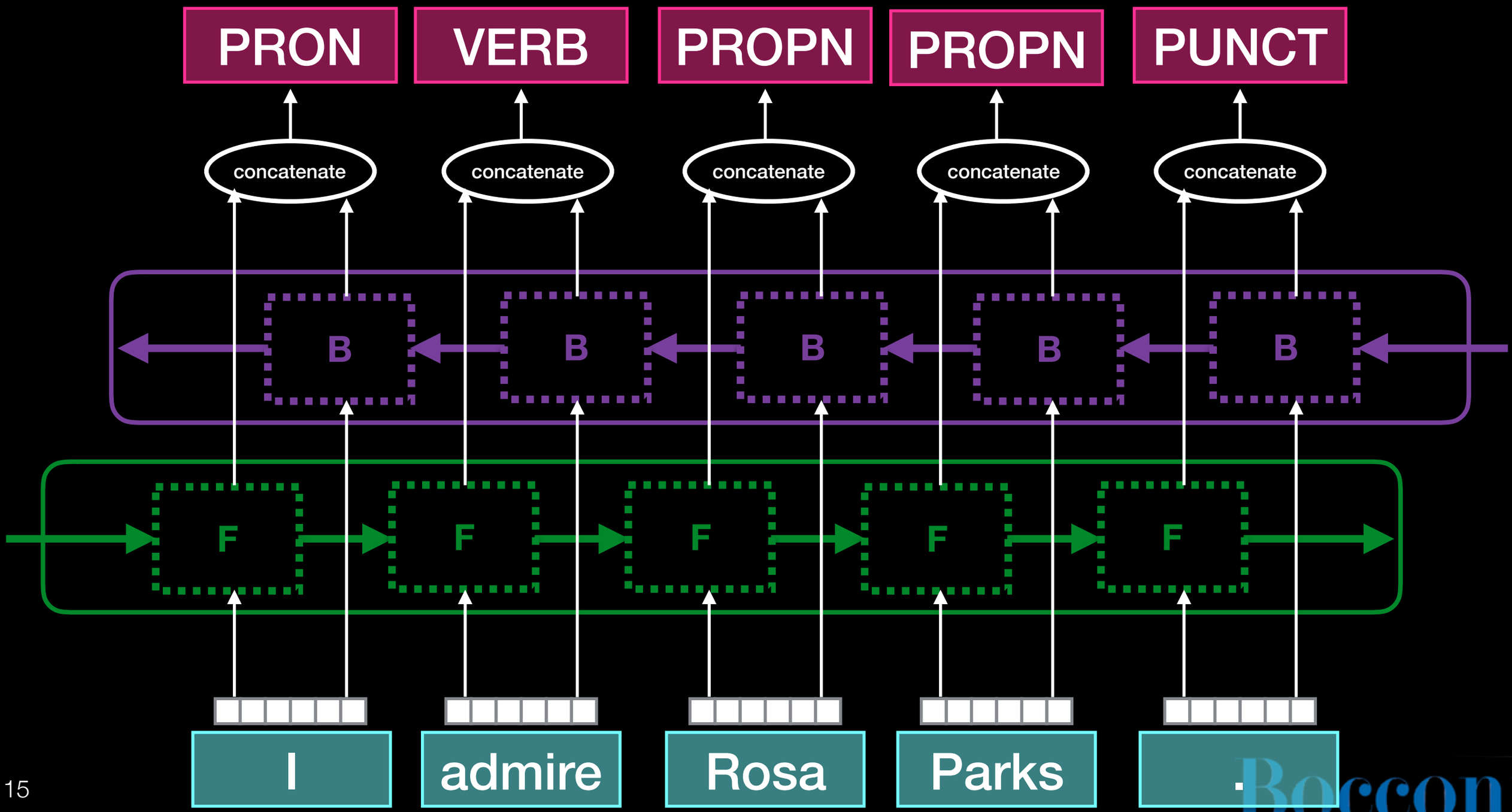
# RNN Tagging

## Structured Prediction



# Bidirectional-RNN

## Structured Prediction



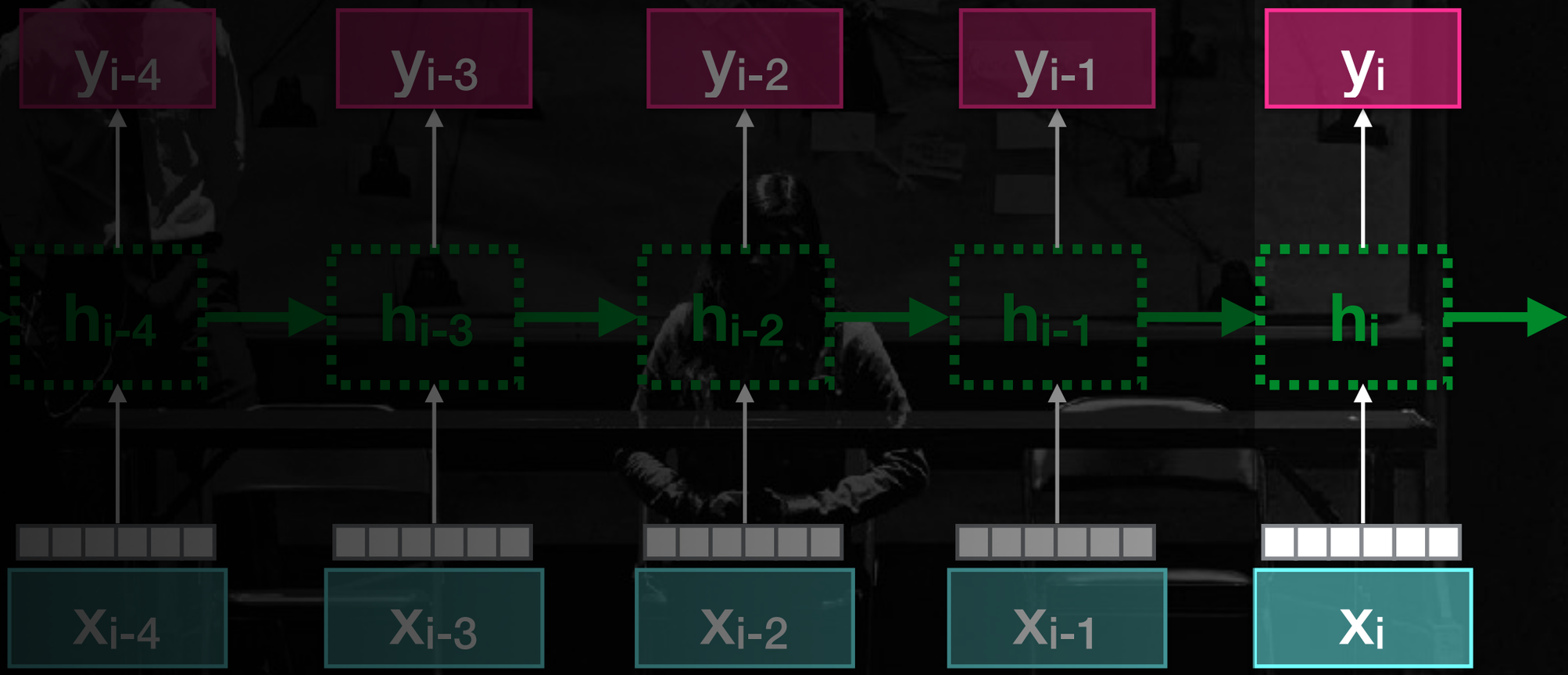
# Special

# Recurrent Networks

Bocconi

# Vanishing Memory

WHERE WERE YOU MARCH 3, 2016?




PROBLEM WITH LONG SEQUENCES

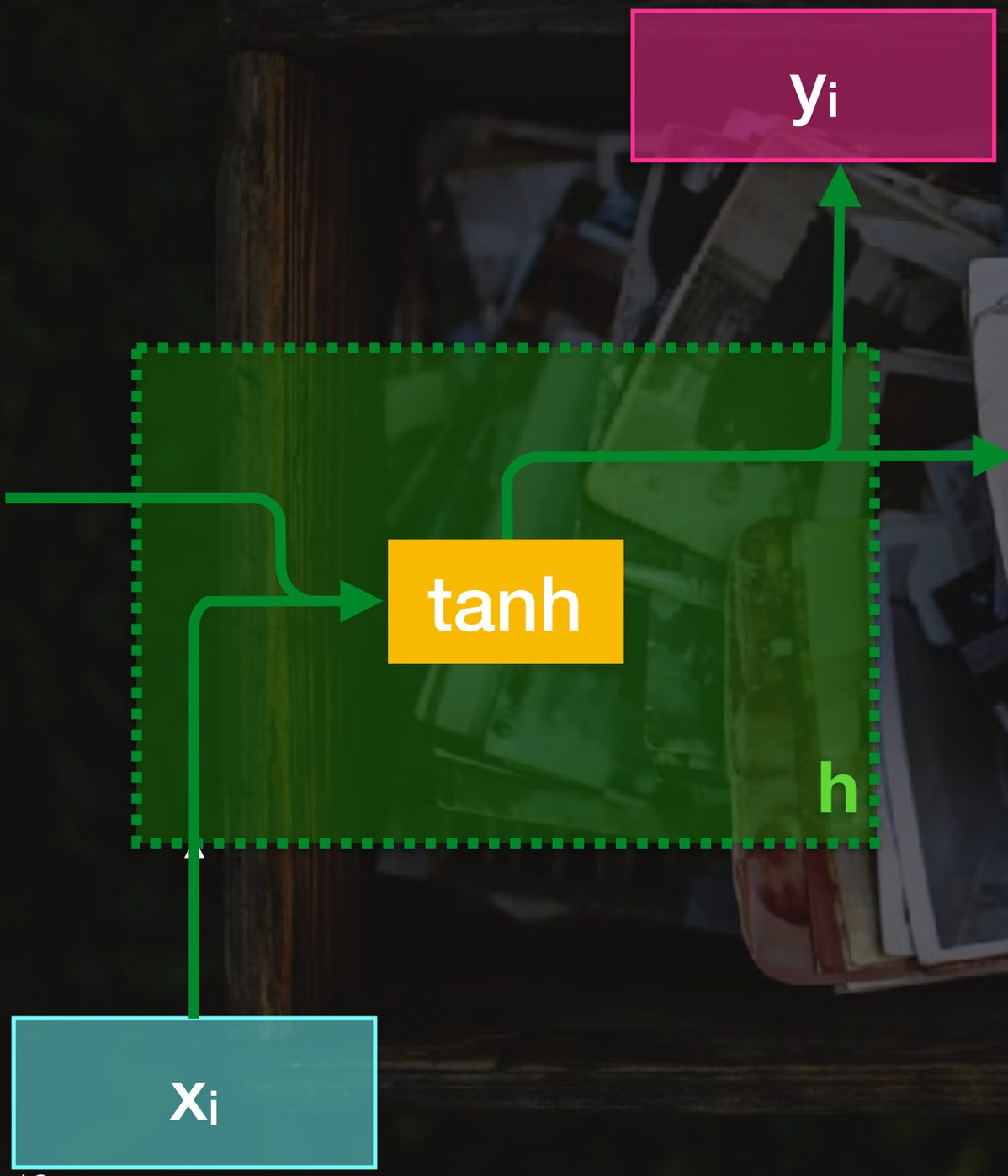


Bocconi

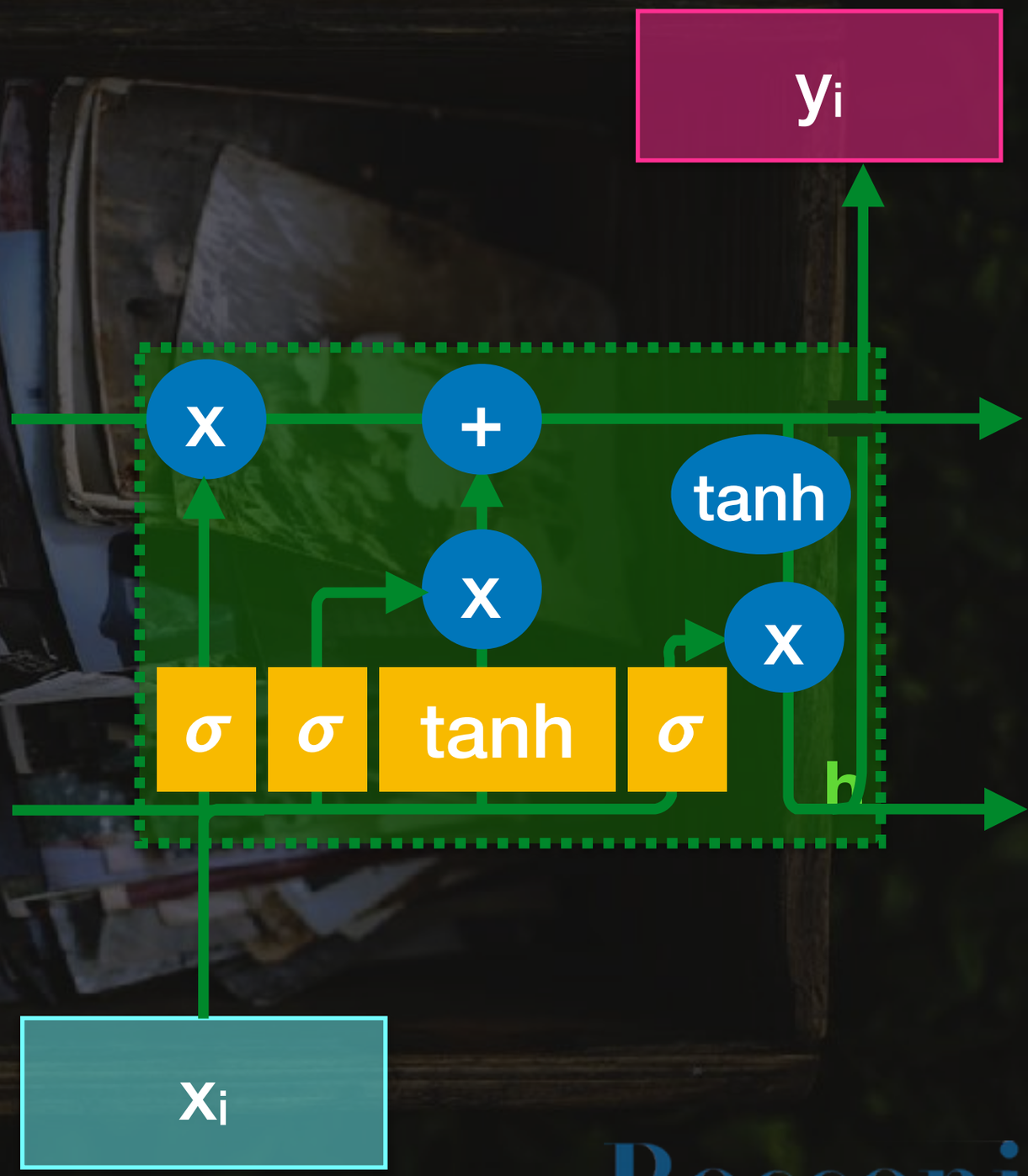
# Selective Forgetting

$$h_i = tanh(W_1 h_{i-1} + W_2 x_i + b)$$

SIMPLE RNN

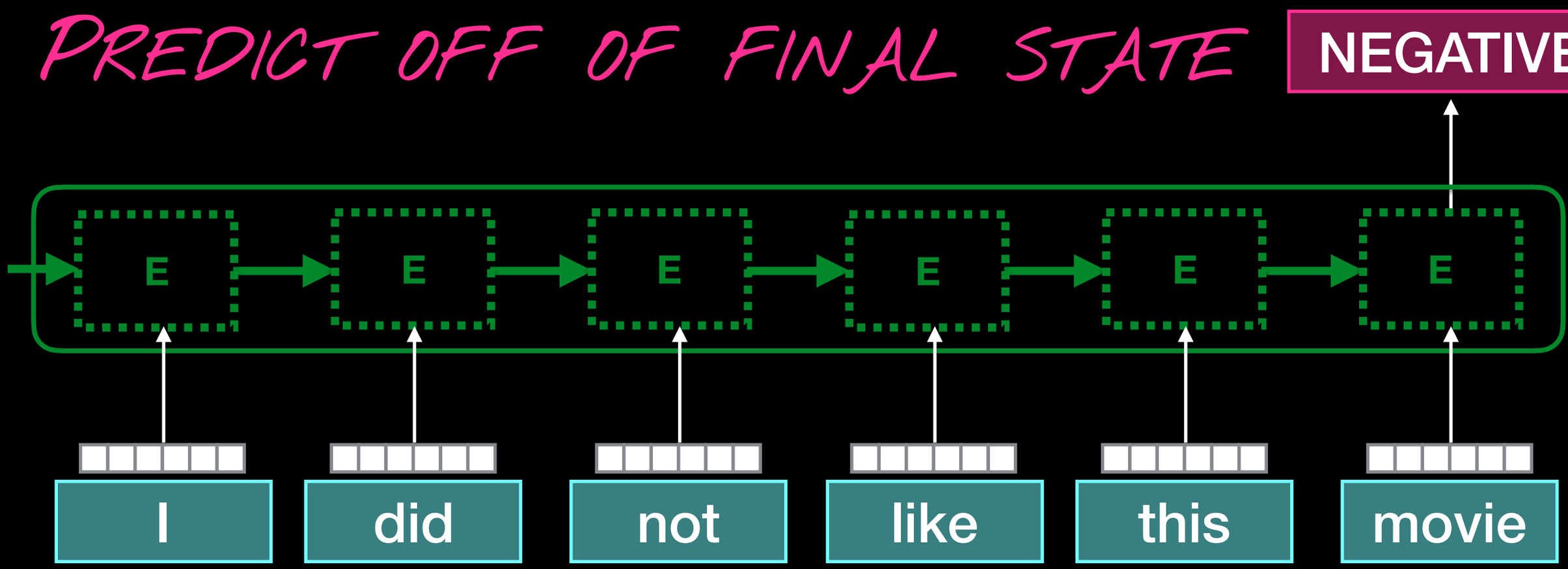



LSTM



# Acceptor

PREDICT OFF OF FINAL STATE

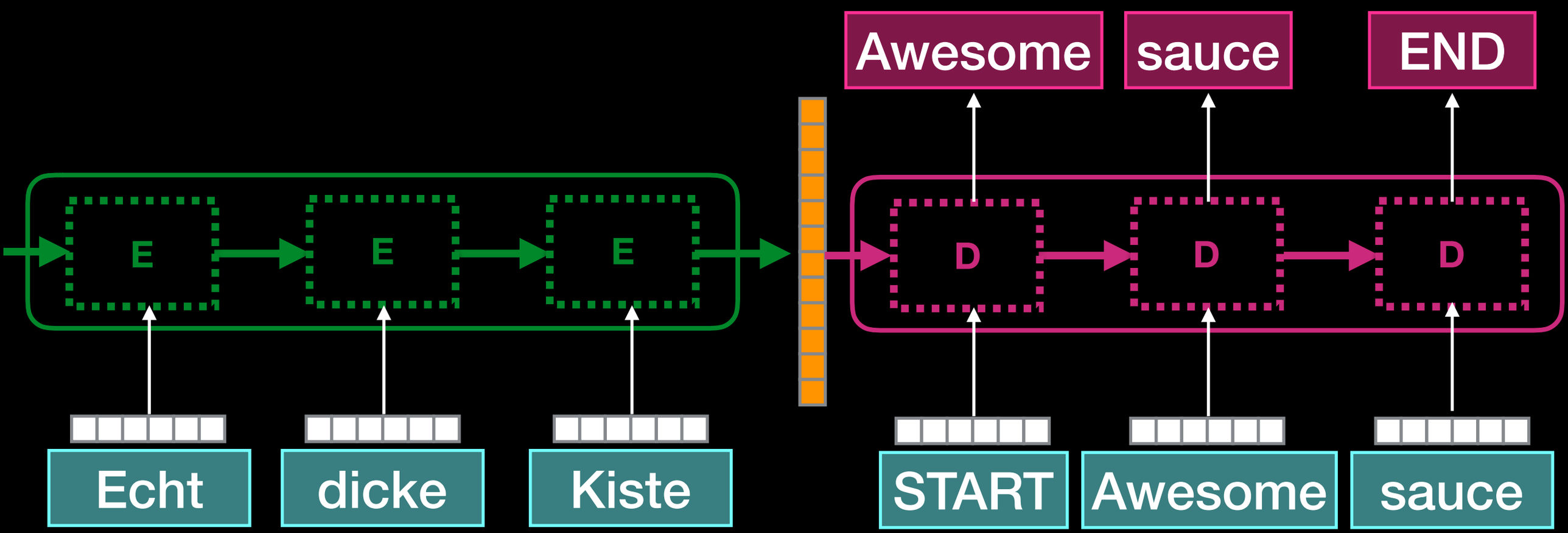





Bocconi

# Encoder-Decoder

…And Generate Output from it




Gobble up sequence into a vector…



Bocconi

# Convolution

Bocconi

# Convoluted Matters

Text Sort classification

Retail

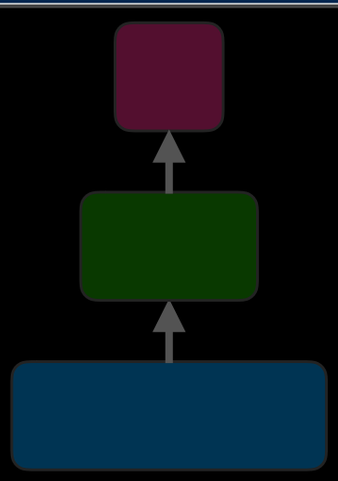
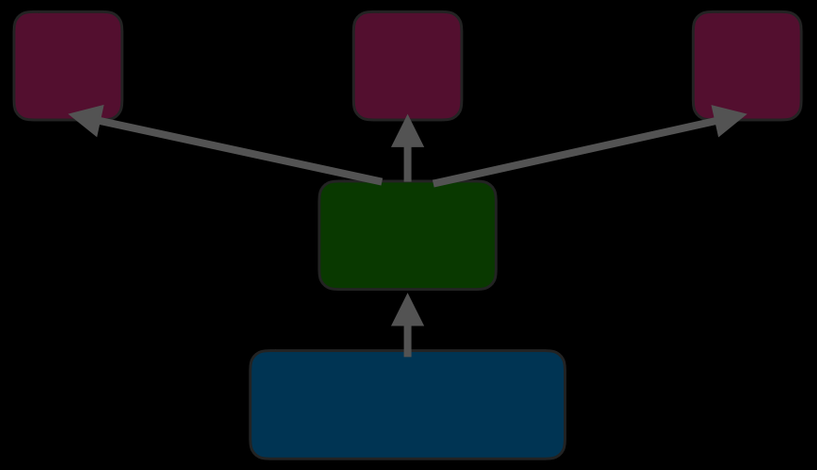
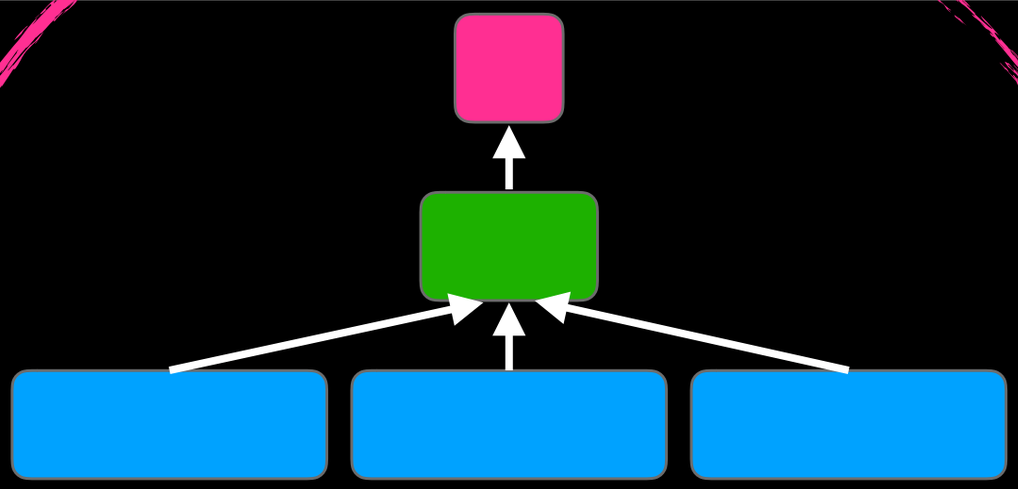
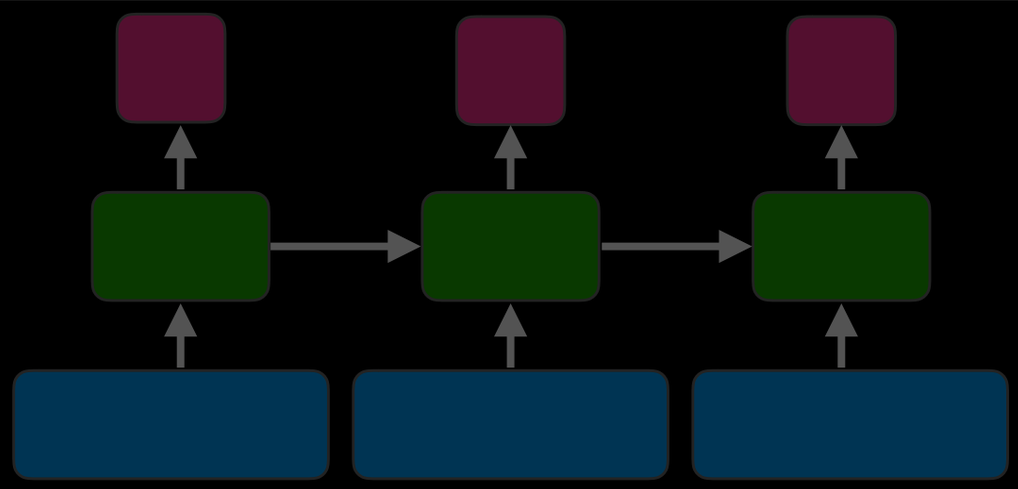
Sentiment Analysis

positive

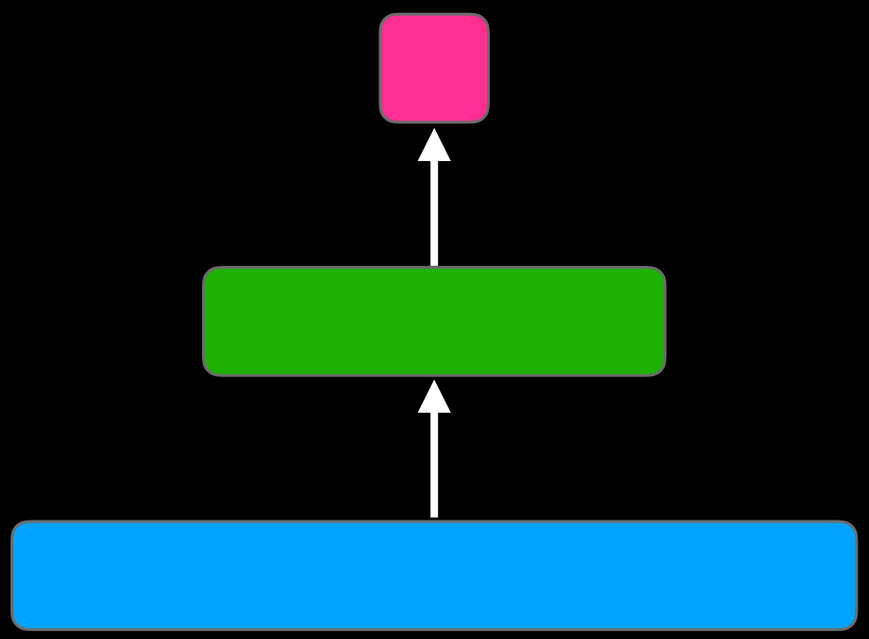
Relation Extraction

founded_by(Amazon, Jeff Bezos)

# Types of Text Classification

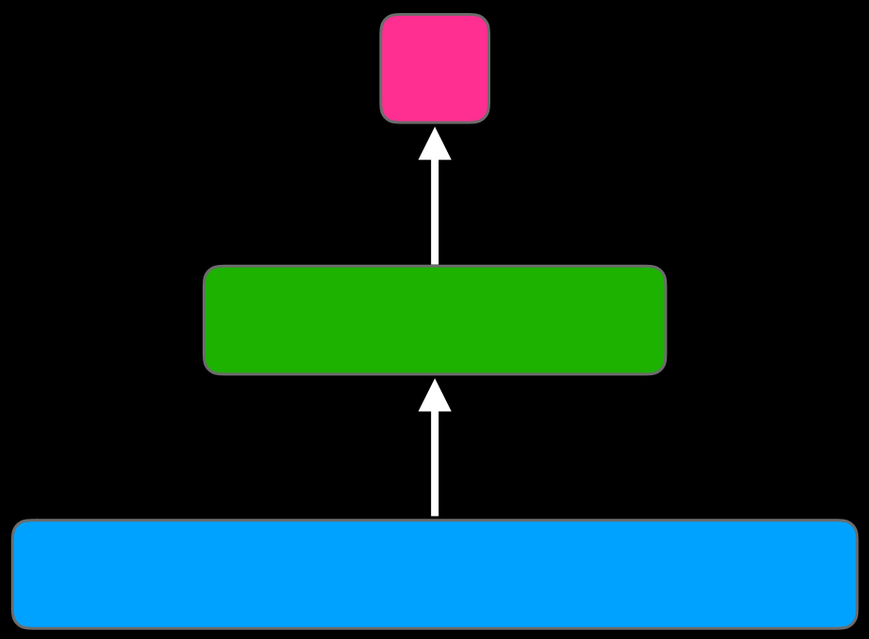
| | Fixed length | Variable length |
|---|---|---|
| **Fixed length** |  Logistic Regression, Perceptron, Feed-Forward Network, Random Forest, Naive Bayes, SVM, … |  Multitask Learning, Decoder |
| **Variable length** |  Convolutional Neural Networks (CNN) |  Recurrent Neural Networks (RNN), Hidden Markov Models (HMM), Conditional Random Fields |

# Problems with MLPs



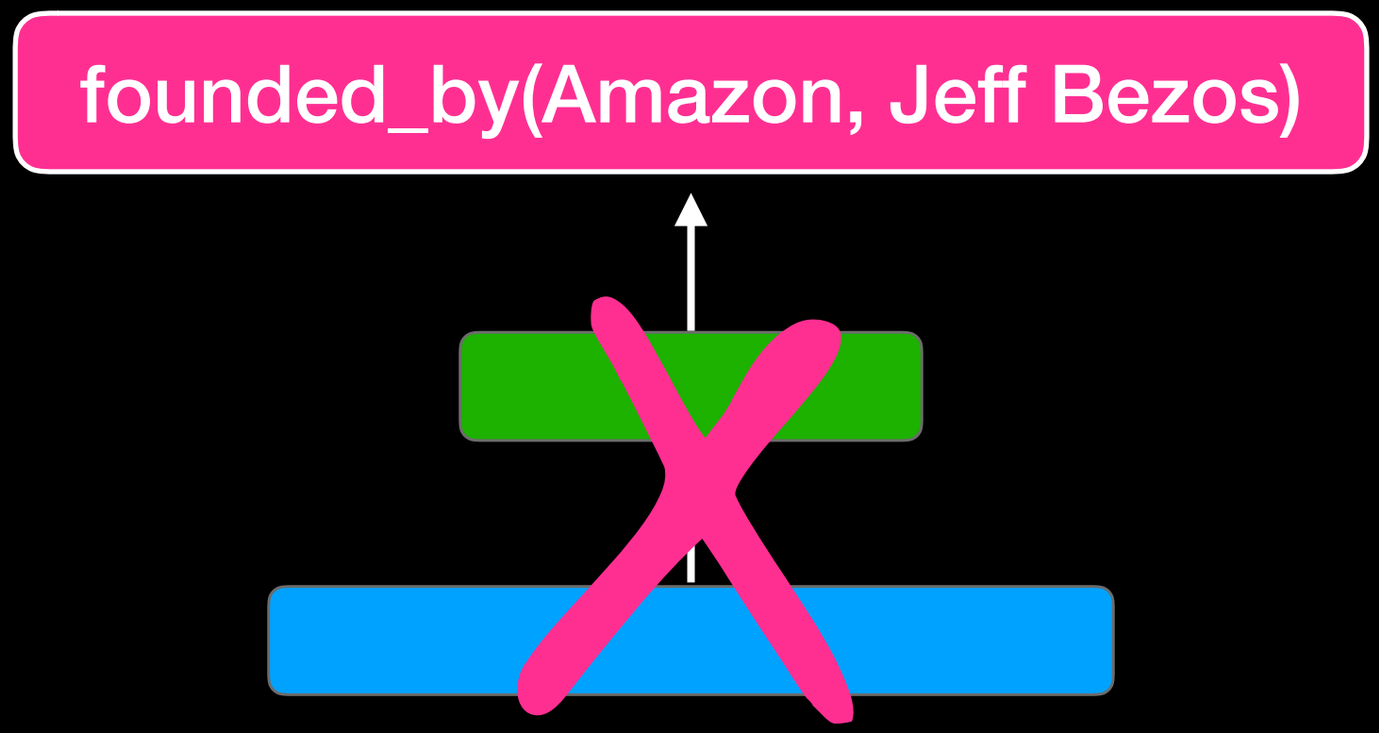
Not bad, actually quite good

$\neq$



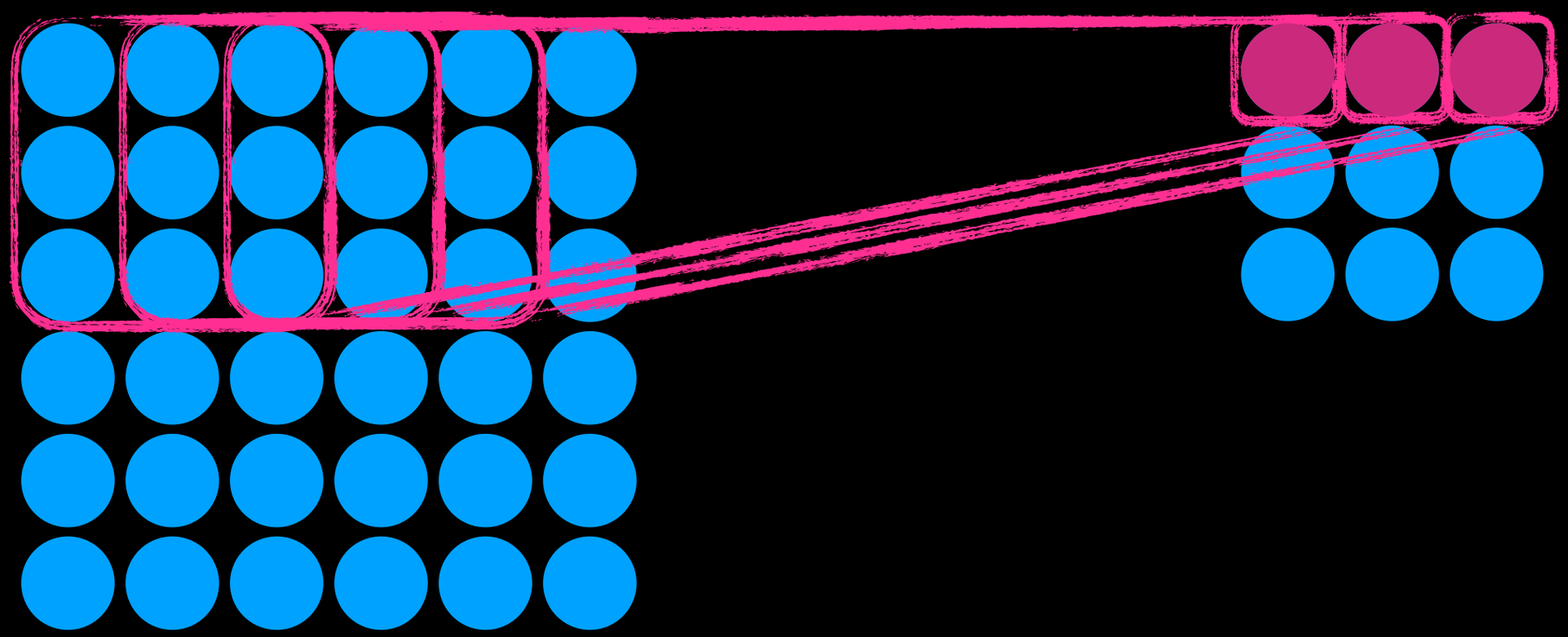
Not good, actually quite bad

# Problems with MLPs

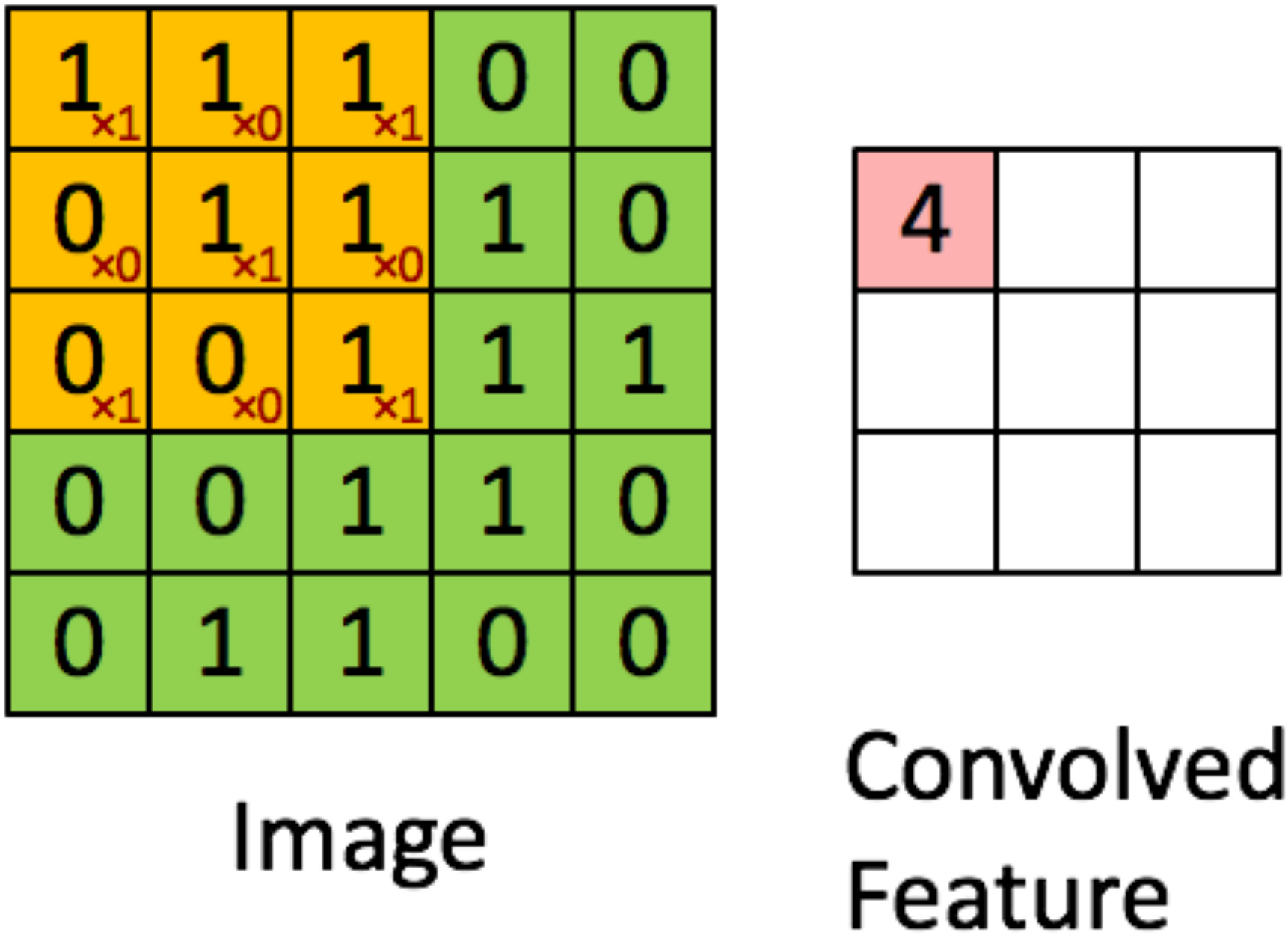



Jeff Bezos, or what Dr. Evil would look like on steroids, went from book seller to billionaire when he founded Amazon in 1994.

# Convolution



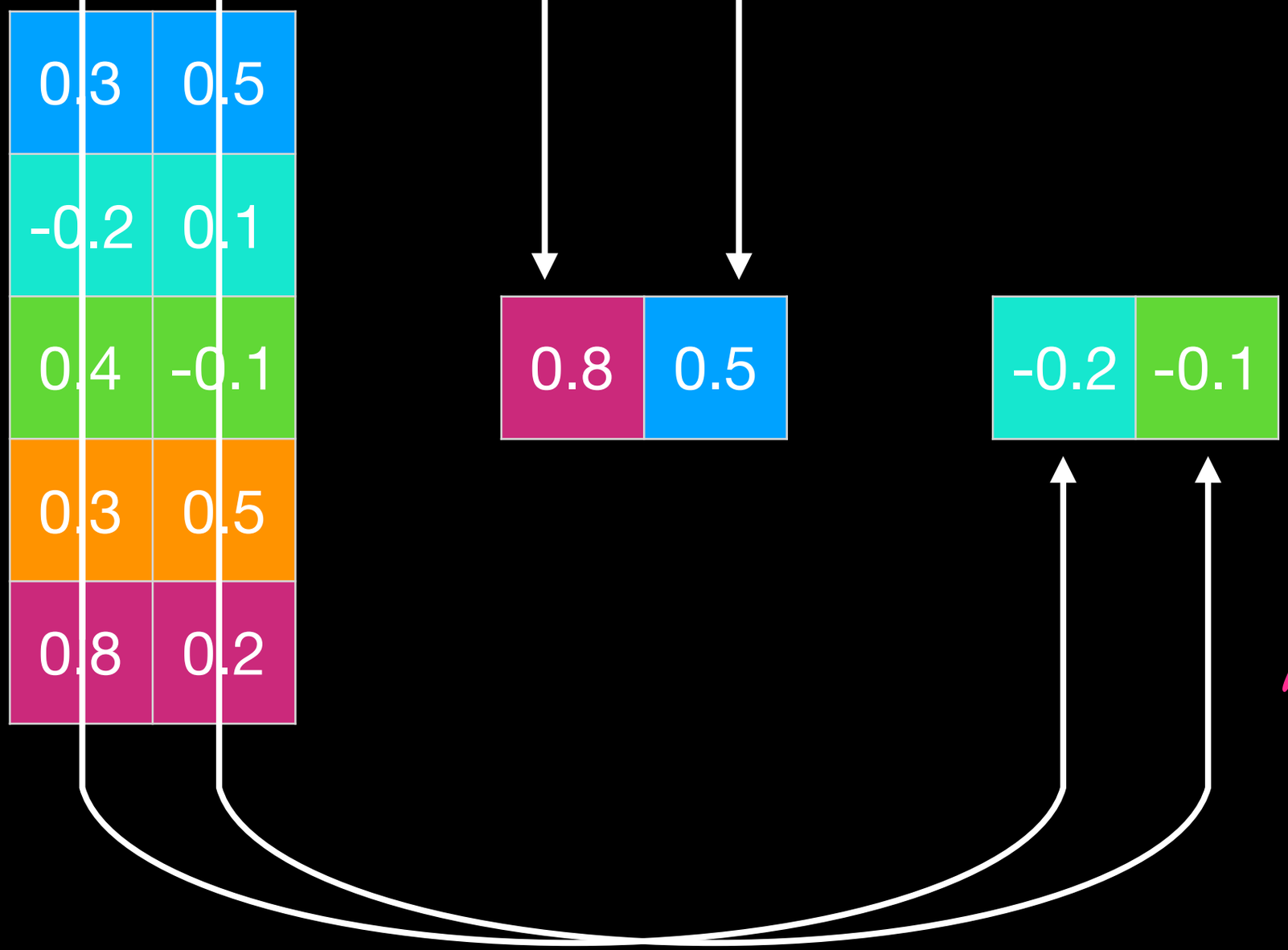
# Convolution Extraction




http://deeplearning.stanford.edu/wiki/index.php/Feature_extraction_using_convolution

# Pooling

MAX POOLING

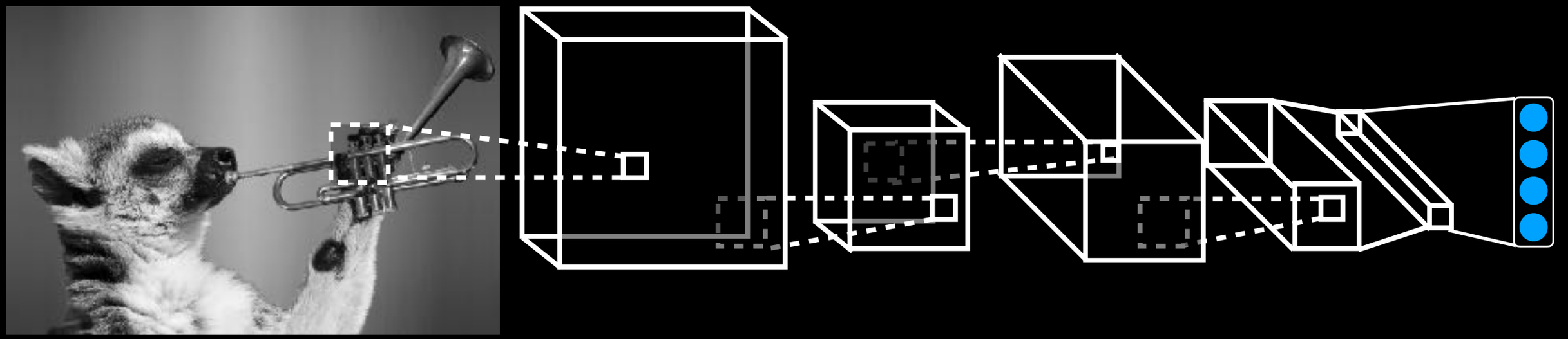



MIN POOLING



Bocconi

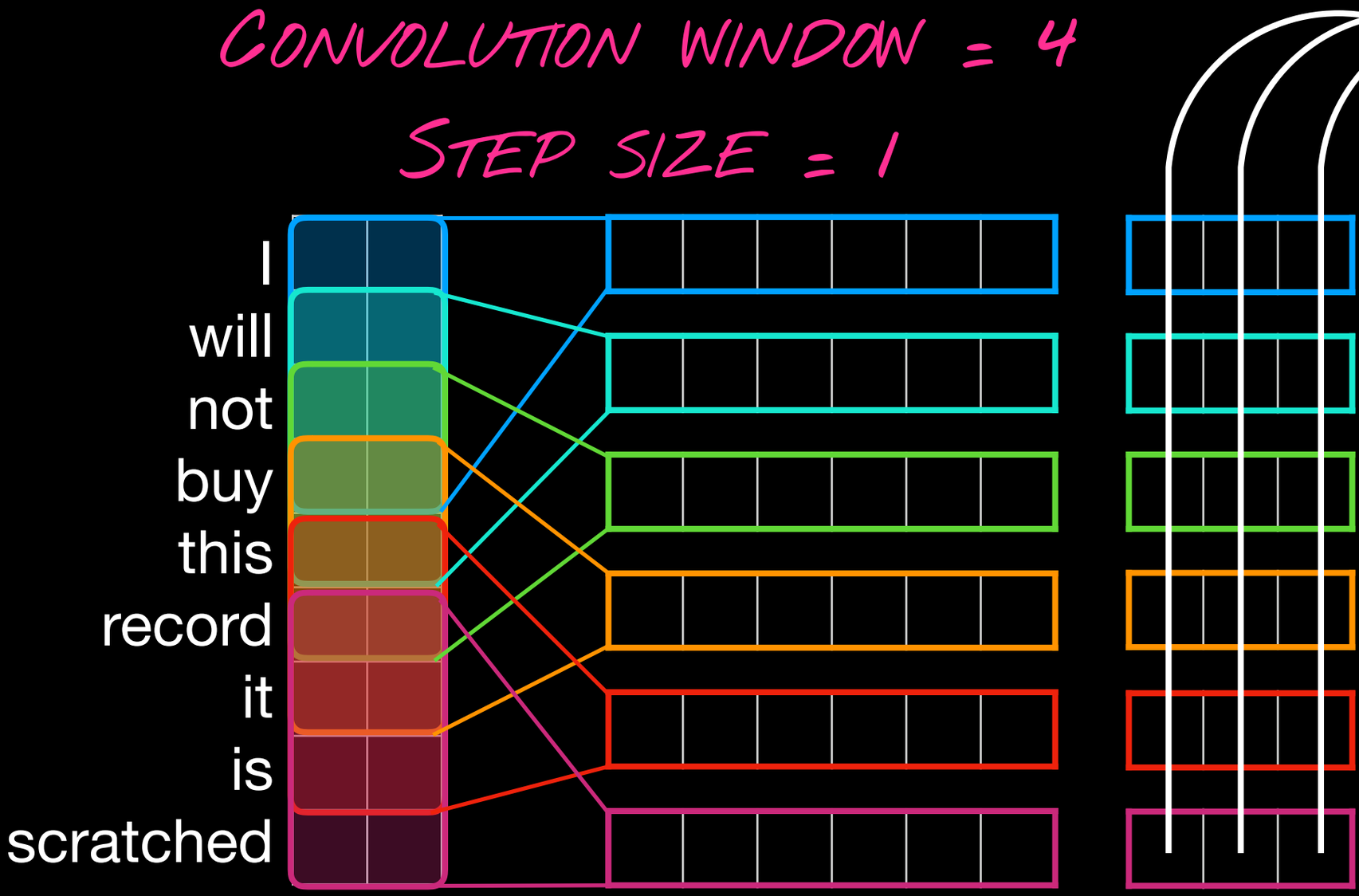
# CNNs in Images

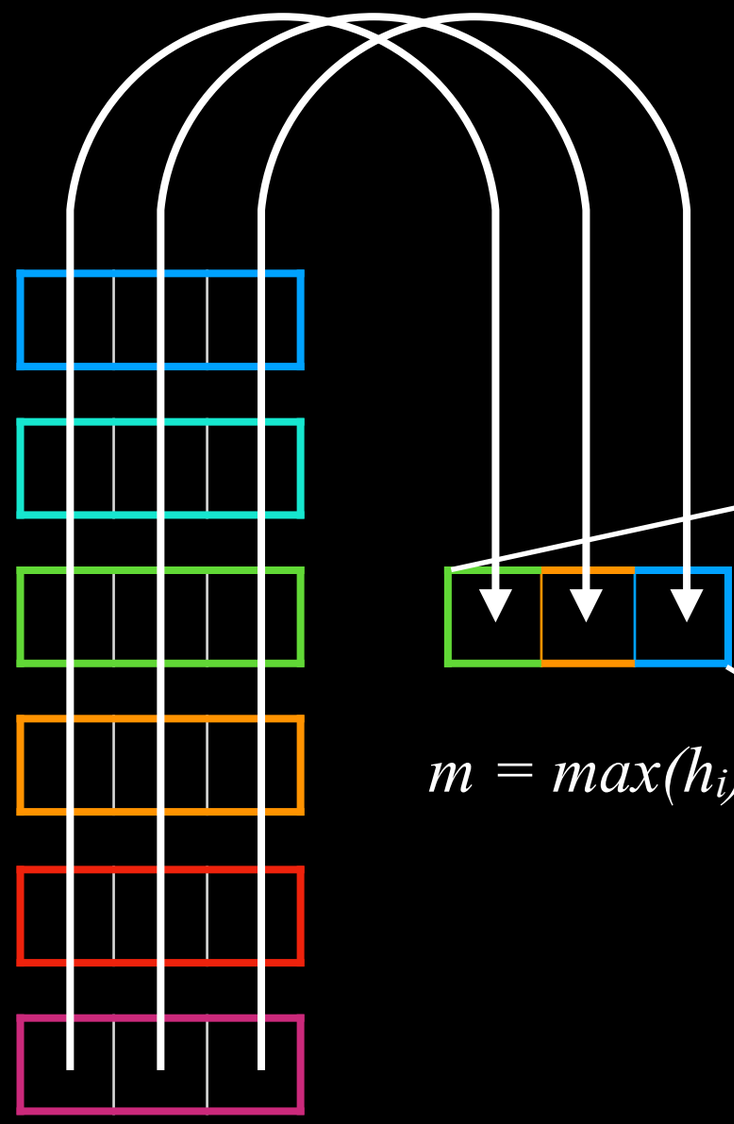


# CNNs in Text

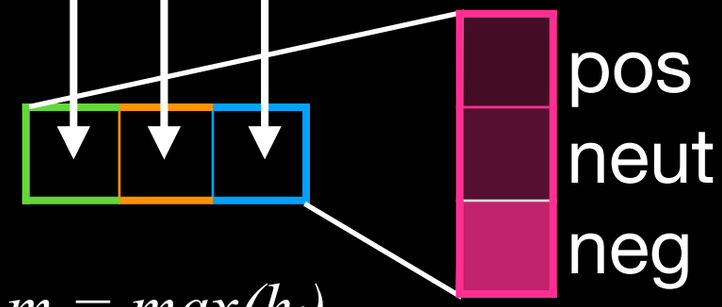
CONVOLUTION WINDOW = 4

STEP SIZE = 1



$y = W_2 \bullet m + b_2$




$m = max(h_i)$

$c_i = concat(x_{i-i+3})$ $\quad h_i = relu(W_1 \bullet c_i + b_1)$



Bocconi

# The Attention Mechanism

Bocconi

# Attention!

- Learn syntactic and semantic relations between words in
  - the input and output (RNNs)
  - only the input (CNNs)
- Good for machine translation (word alignment) and classification (complex expressions)
- Basis of the Transformer model

# Visual Attention

Oh look, a furry ear!

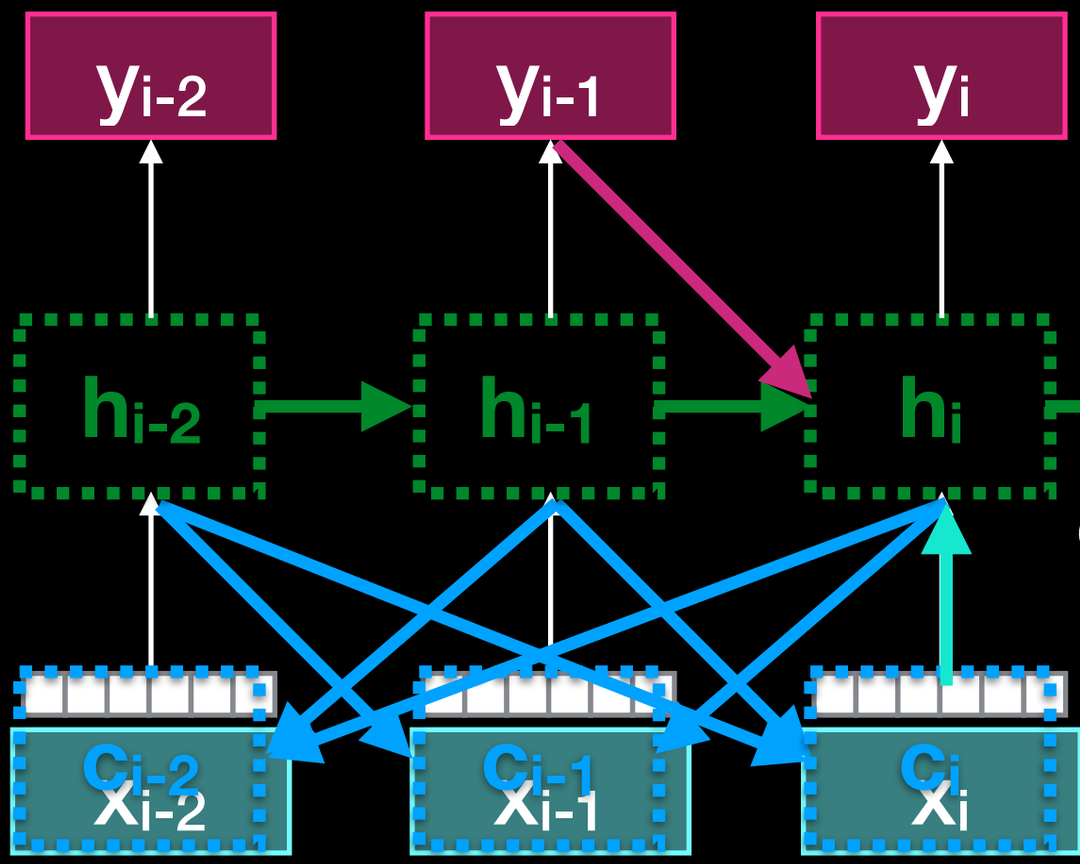
Of course, there's a doge in the picture:
look at all the other doge parts

Ignore those other parts, they don't tell you anything about doges

see https://lilianweng.github.io/lil-log/2018/06/24/attention-attention.html

# Ok, but how?

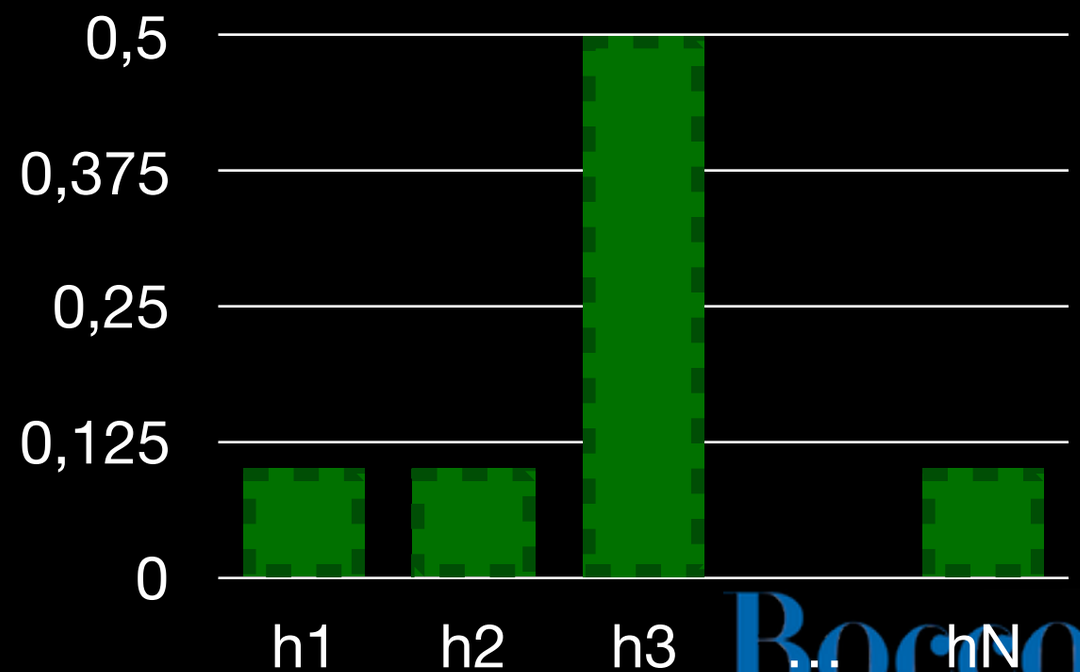
$$h_i = f(h_{i-1}, y_{i-1}, c_i)$$




$$c_i = \sum_{j=1}^{N} \alpha_{i,j} h_j$$

ATTENTION WEIGHTS

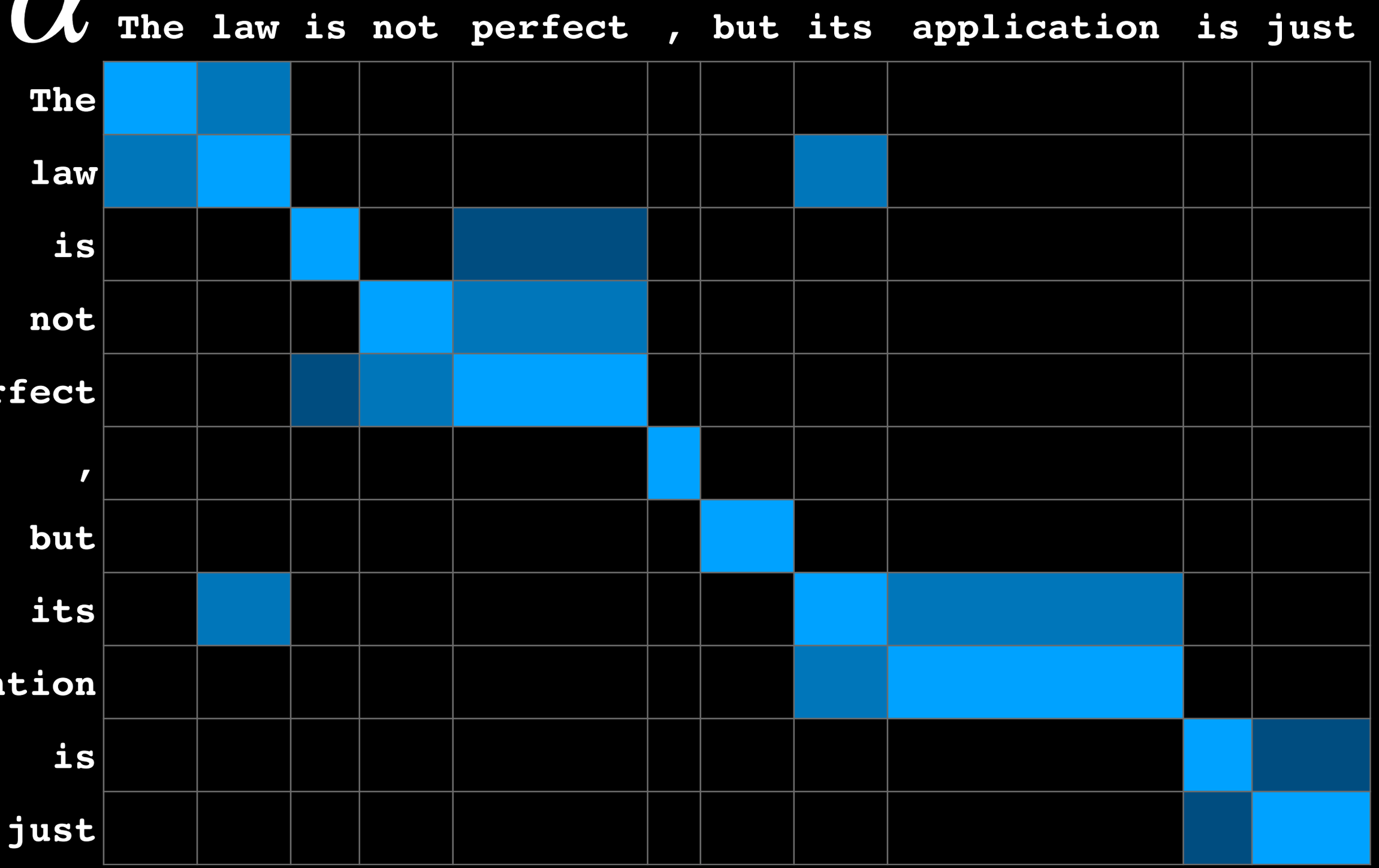
$$\alpha_{i,j} = softmax(v^T \tanh(W[s_i; h_j]))$$



# But Why?

- RNNs can't store long sequences
- CNNs don't have a sense of location in the final vector
- Attention tells us how each part relates to each other part

# CNN with Attention




FIND LONG-RANGE DEPENDENCIES



Bocconi

# RNN with Attention

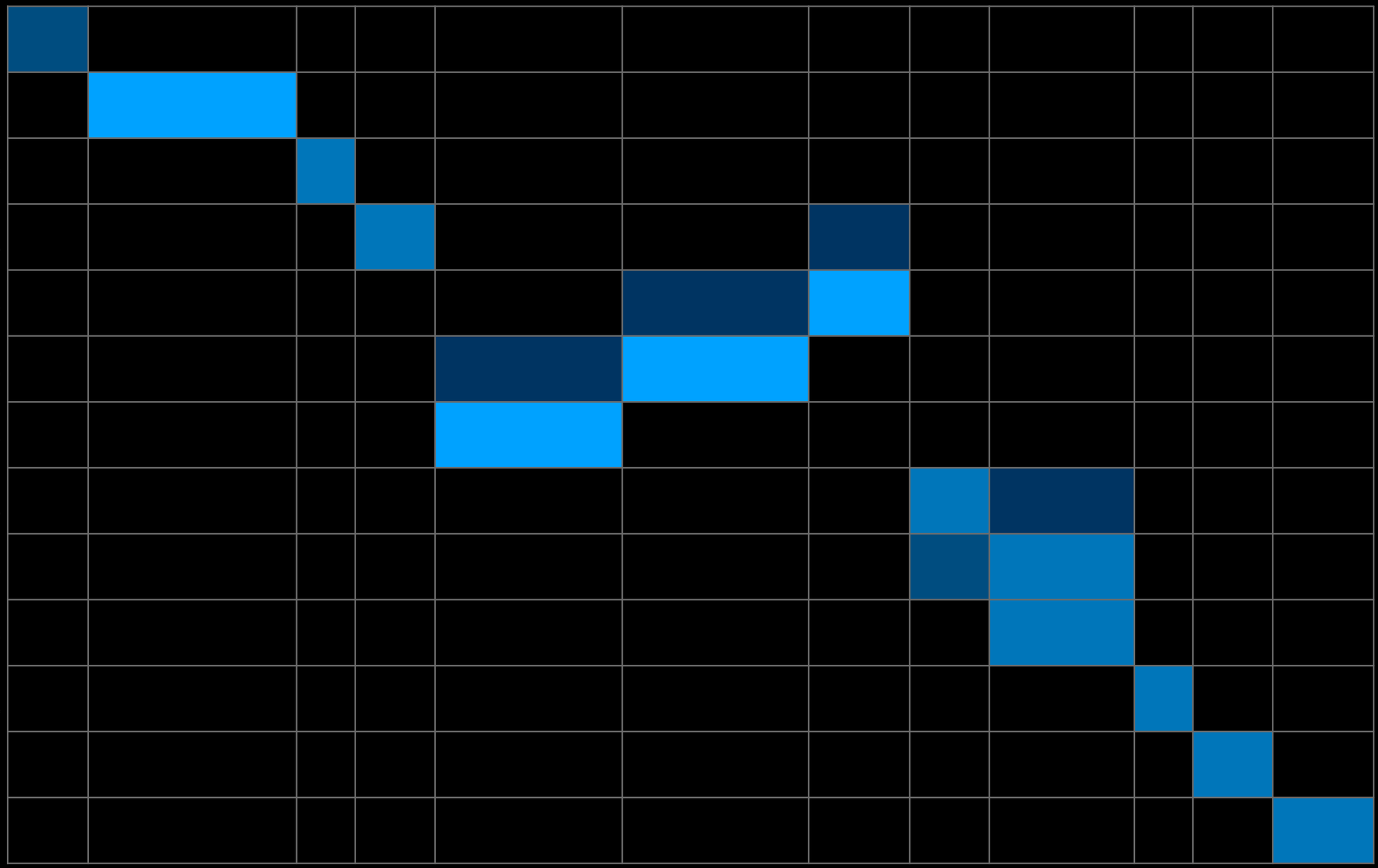
OUTPUT

$\alpha$

The agreement on the European Economic Area was signed in Aug 1992

INPUT

L'
accord
sur
la
zone
économique
européenne
a
été
signé
on
août
1992



LEARN REORDERING



Bocconi

# RNN with Attention


LEARN REORDERING



Bocconi

# Multimodal Attention

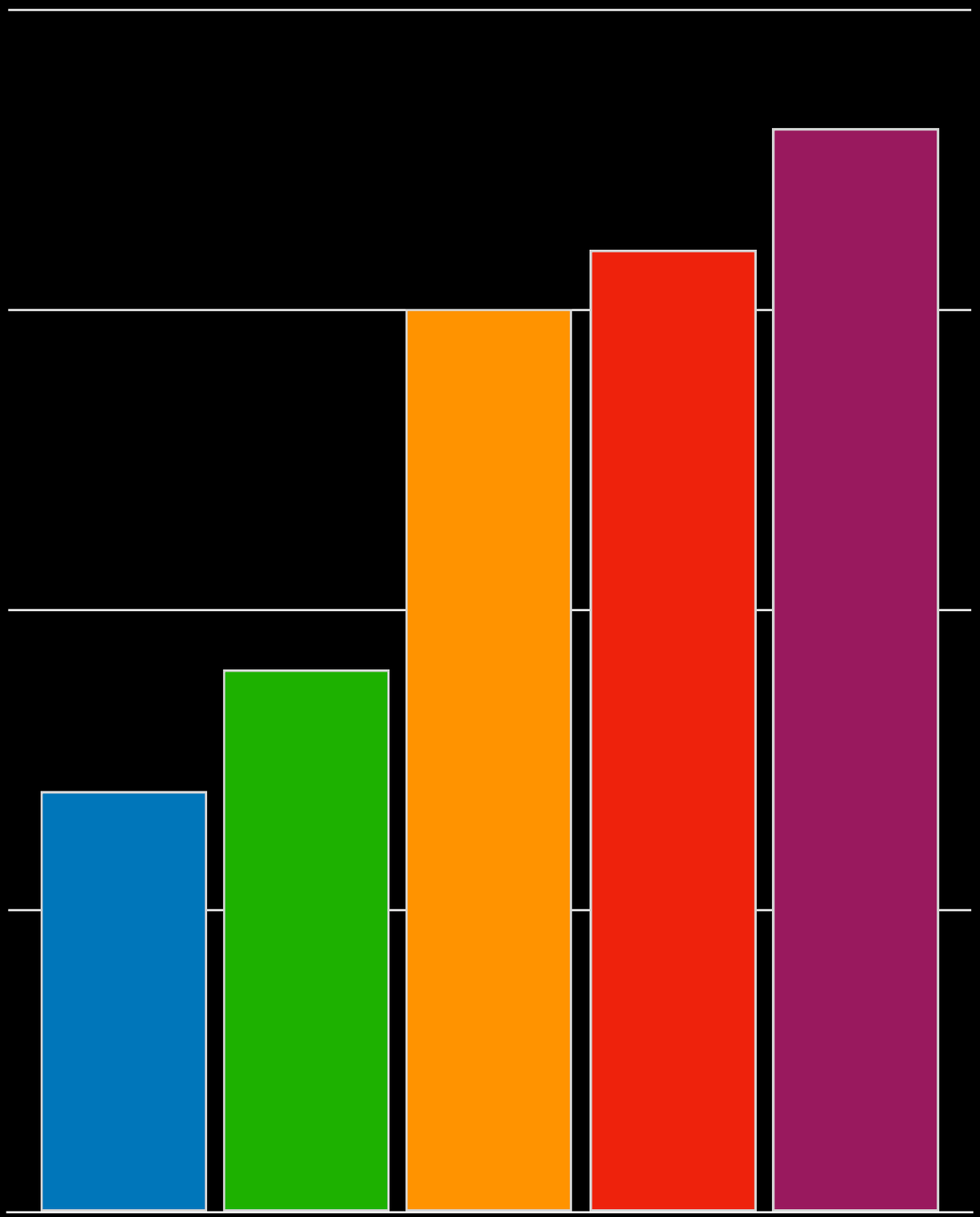
**A woman is throwing a frisbee in the park .**



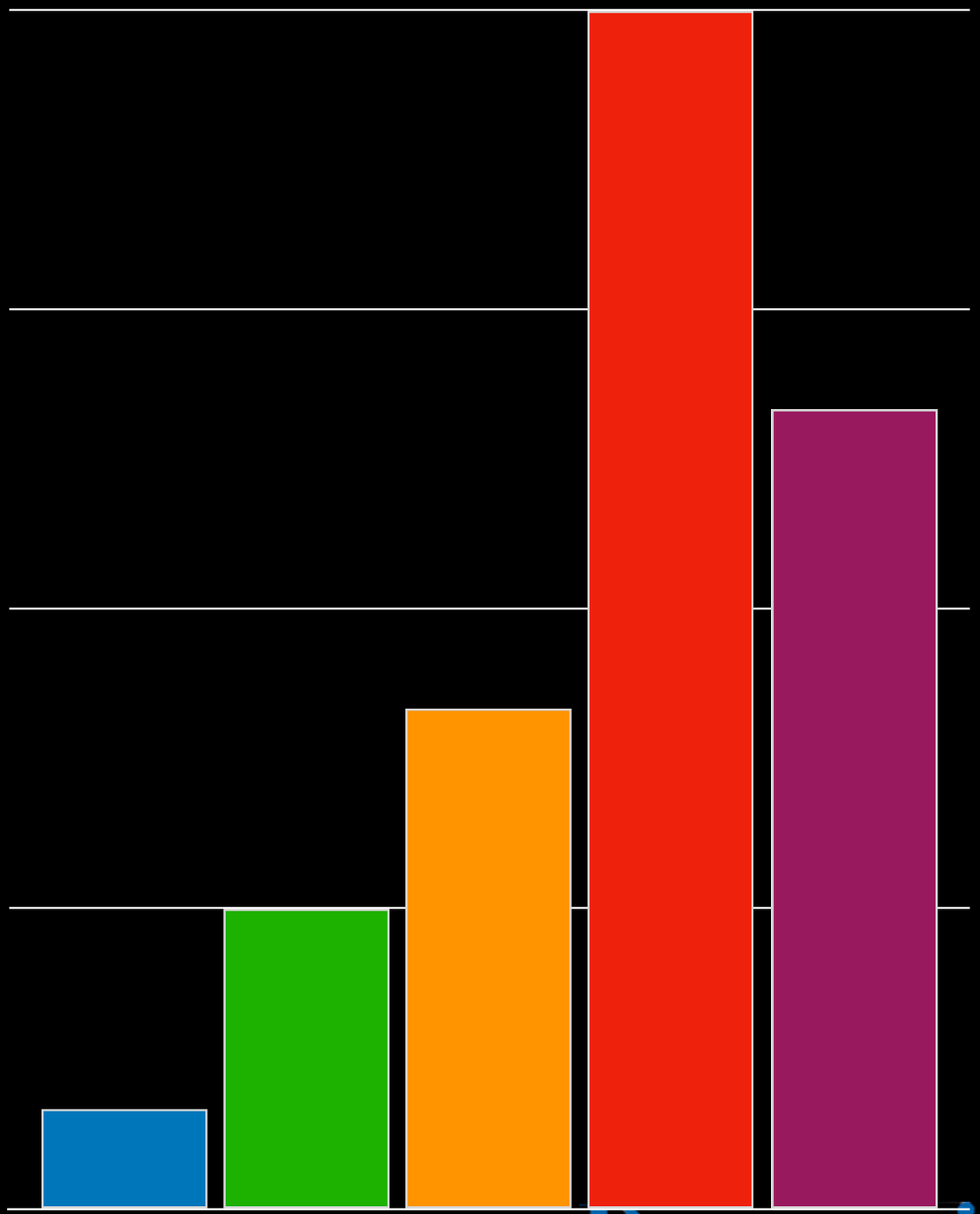

Bocconi

# Performance vs Parameters

Logistic Regression | Feed-Forward | CNN | RNN | CNN+Attention



performance



parameters

# Wrapping up

Bocconi

# Take Home Points

- **Recurrent Neural Nets** address long-range dependencies
- Condition each word on all previous ones (better for **LMs** and **sequence labels)**
- **Bidirectional RNNs** condition on following words
- **LSTMs** learn to forget useless input
- **Convolution windows** captures **different views** of input
- **Pooling** reduces dimensionality
- CNNs are often **better for text classification** than feedforward NNs
- **Attention** improves coherence and performance