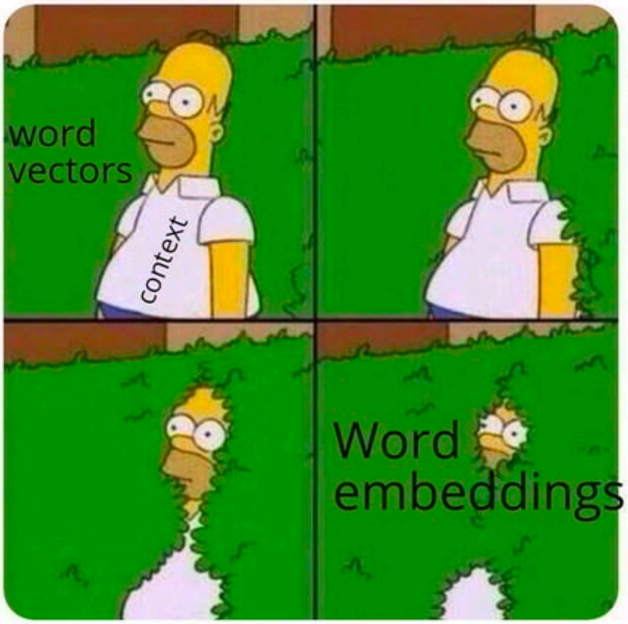
May 3, 2023.

DSML : NLP module.

Recurrent Neural Networks.

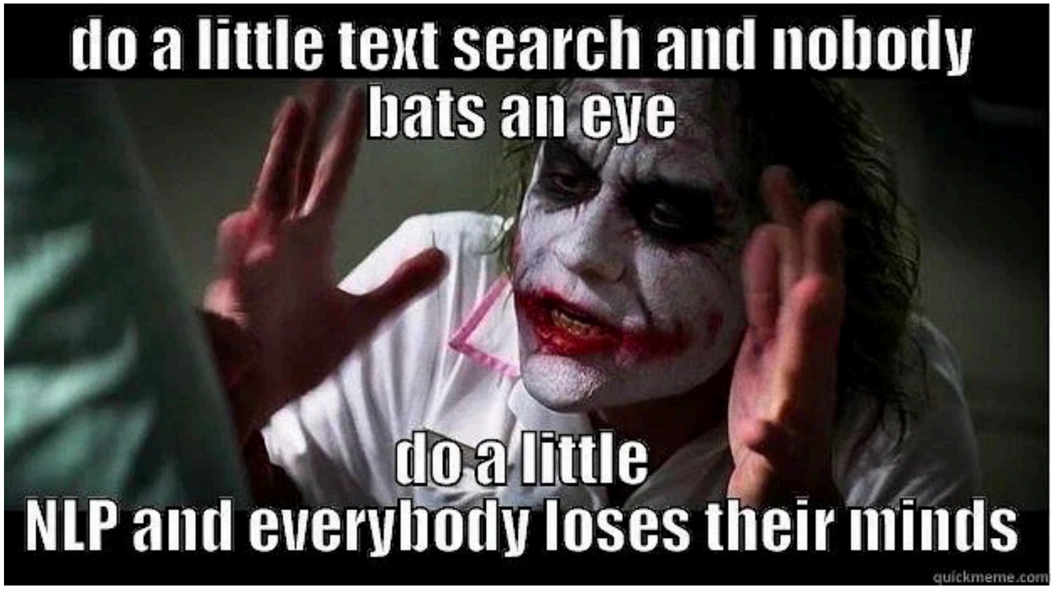
Class starts @ 9:05

Word embeddings in a nutshell



When you penalize your Natural Language Generation model for large sentence lengths

Me think, why waste time say lot word, when few word do trick.


do a little text search and nobody bats an eye
do a little NLP and everybody loses their minds
quickmeme.com

Recap:

- * Document Vectorization: Bag of Words (BoW)
  TF/IDF.

- * Word Vectorization: Continuous BoW,
  Skip-gram.

- * Language Modeling: Naive Bayes to predict
  the next word.

- * Topic Modeling: Heuristics,
  Latent Dirichlet Allocation

Agenda:

* Recurrent Neural Networks. (RNNs)
* A new type of Architecture.
* Different variants.
* Training - Backpropagation through Time.

But first, business case: Classifying News Articles.

Sequence Data :

- * I travelled to France last December.

  1 2 3 4 5 6

  t = 1 t = 2 t = 4

- * A CV conference was held at Nice.

  7 8 9 10 11 12 13

- * It lasted for three days.

  15 16 17 18 19.

Approaches:

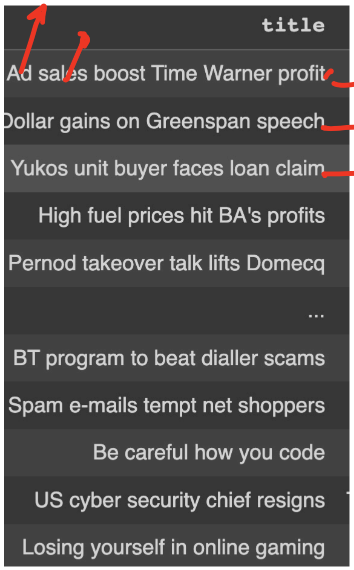
1] Doc → TF/IDF → Multiclass classification.

SVMs, Decision trees,

2] Neural network approaches:

Problem 1] length of inputs is varying.

Solution: clip the input.



title
Ad sales boost Time Warner profit
Dollar gains on Greenspan speech
Yukos unit buyer faces loan claim
High fuel prices hit BA's profits
Pernod takeover talk lifts Domecq
...
BT program to beat dialler scams
Spam e-mails tempt net shoppers
Be careful how you code
US cyber security chief resigns
Losing yourself in online gaming

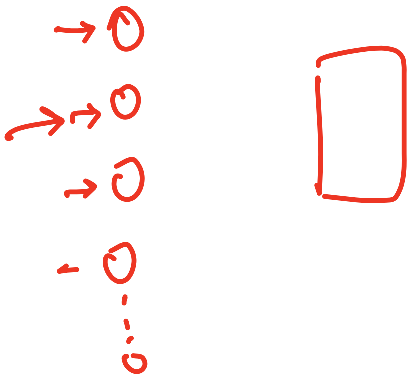
→ 7

→ 5

→ 6

<blank> <blank>

max_length: 12 words.



Problem 2] Context information.

Solution: N-grams. Force the NN to see N words at a time.

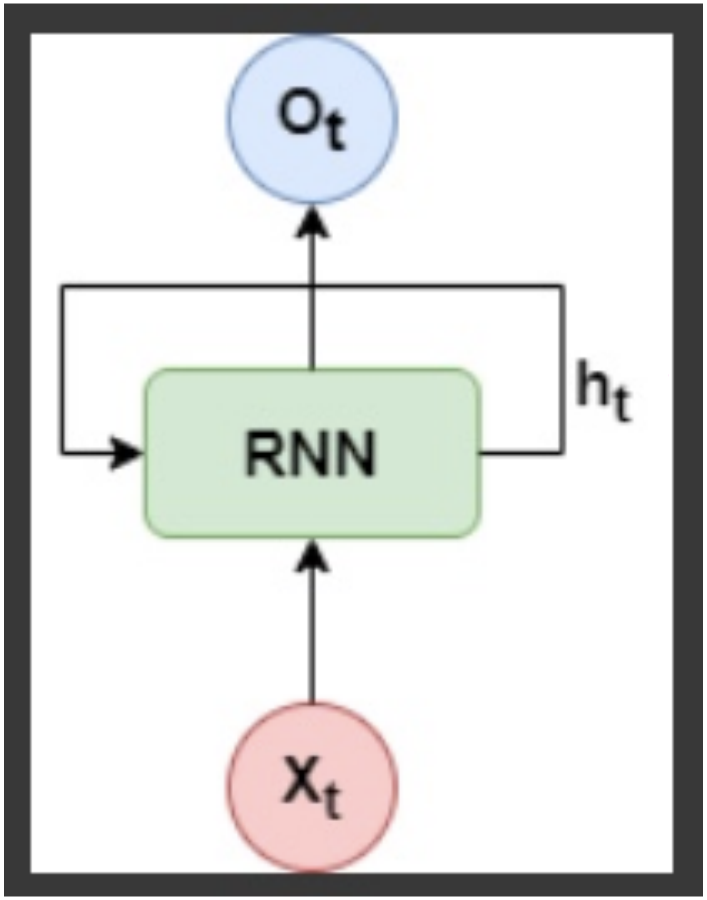
Length of the N-gram is very difficult to optimize.

Problem 3]. Time & space complexity.

Motivation behind this architecture:

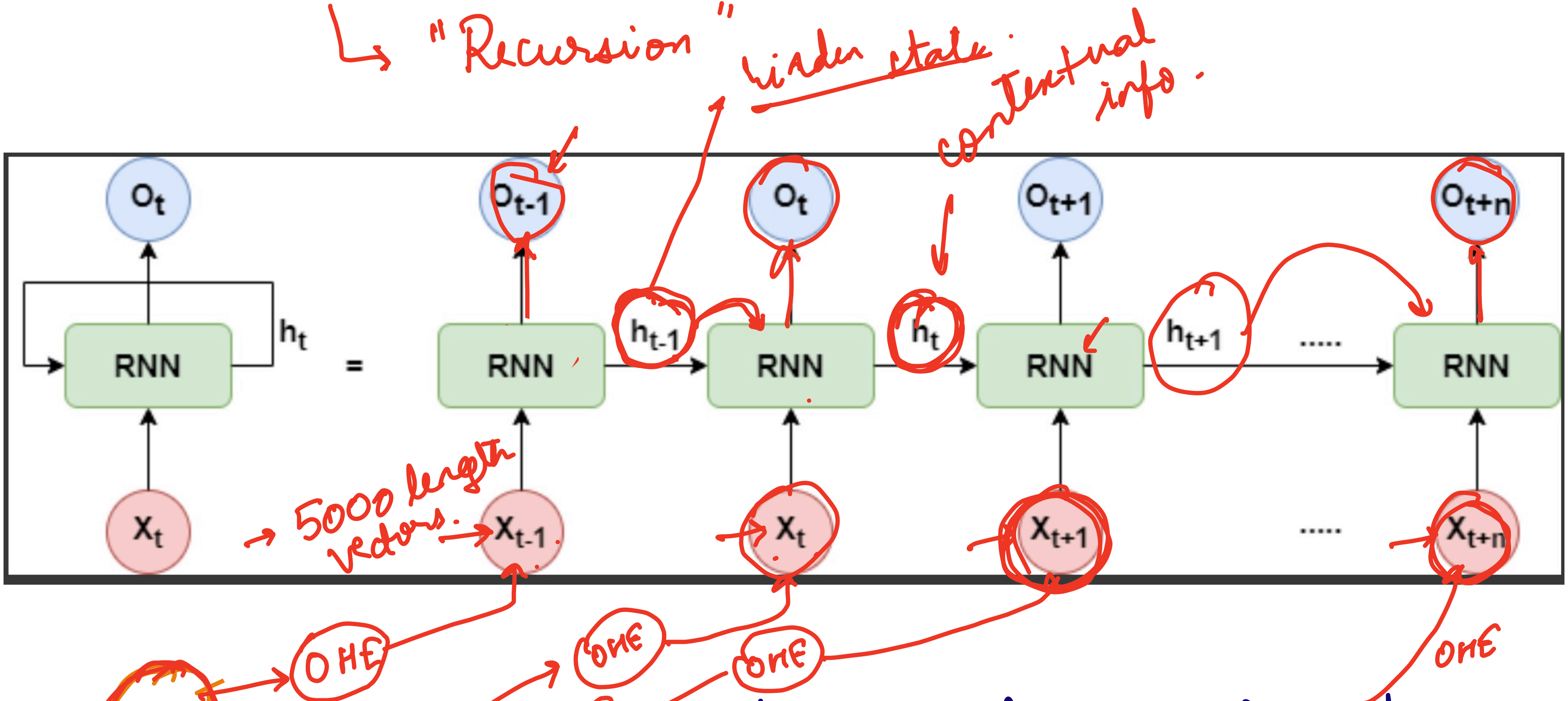
→ NN lack memory.

→ How do we give it memory?



Recurrent Neural Networks.

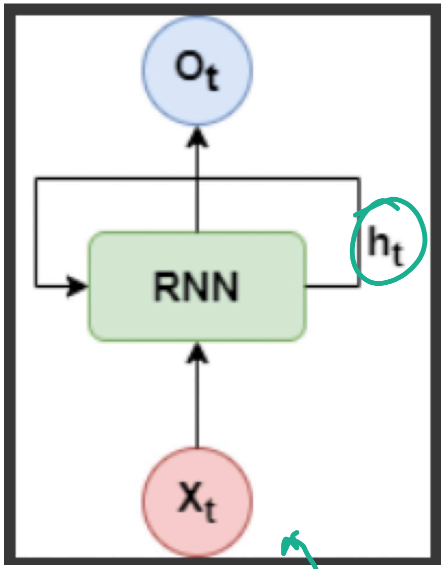
↳ "Recursion"



I travelled to France last December.

1 2 3 4 5 6.

|V| = 5000



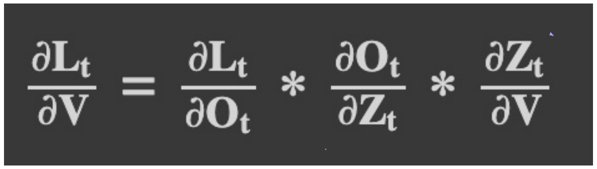
- Garfield is a cat

1 2 3 4

He is fat.

1 2 3

RNN architecture deep dive.



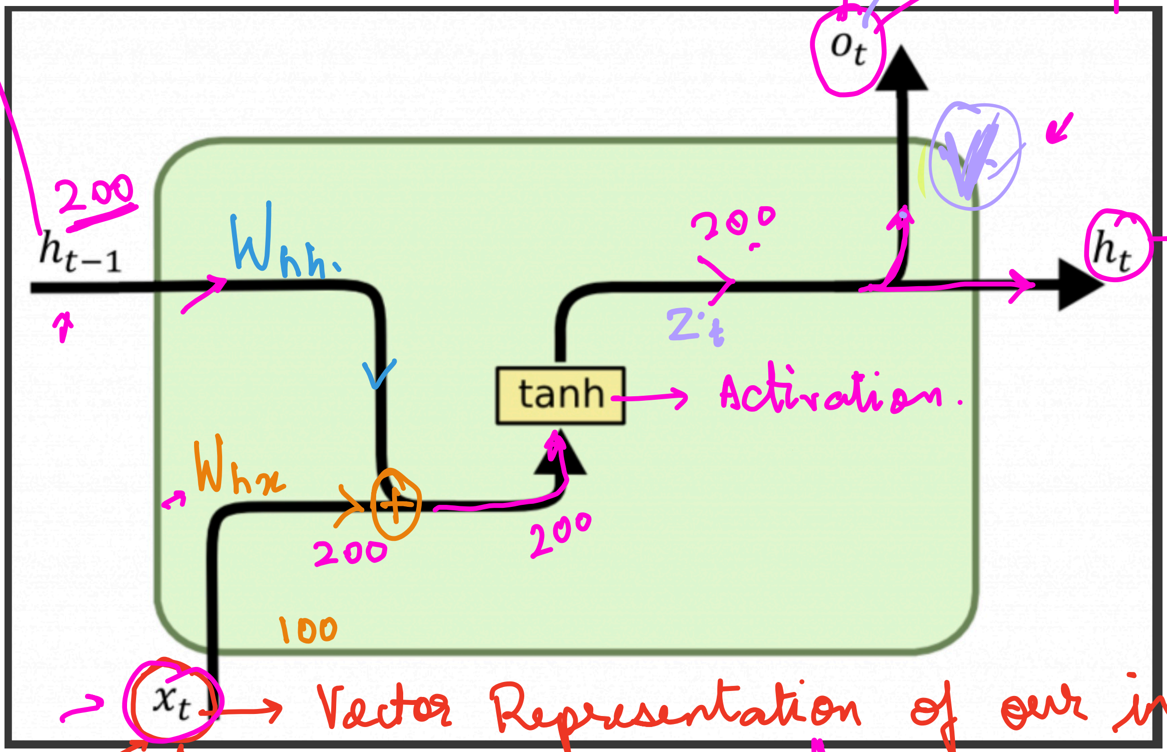
$$\frac{\partial L_t}{\partial V} = \frac{\partial L_t}{\partial O_t} * \frac{\partial O_t}{\partial Z_t} * \frac{\partial Z_t}{\partial V}$$

t = 0.

① h_0 = 0 vector

② h_0 → random.

③ h_0 - some fixed vector.



100

$\frac{\partial L}{\partial O_t}$

output at time t.

→ Activation.

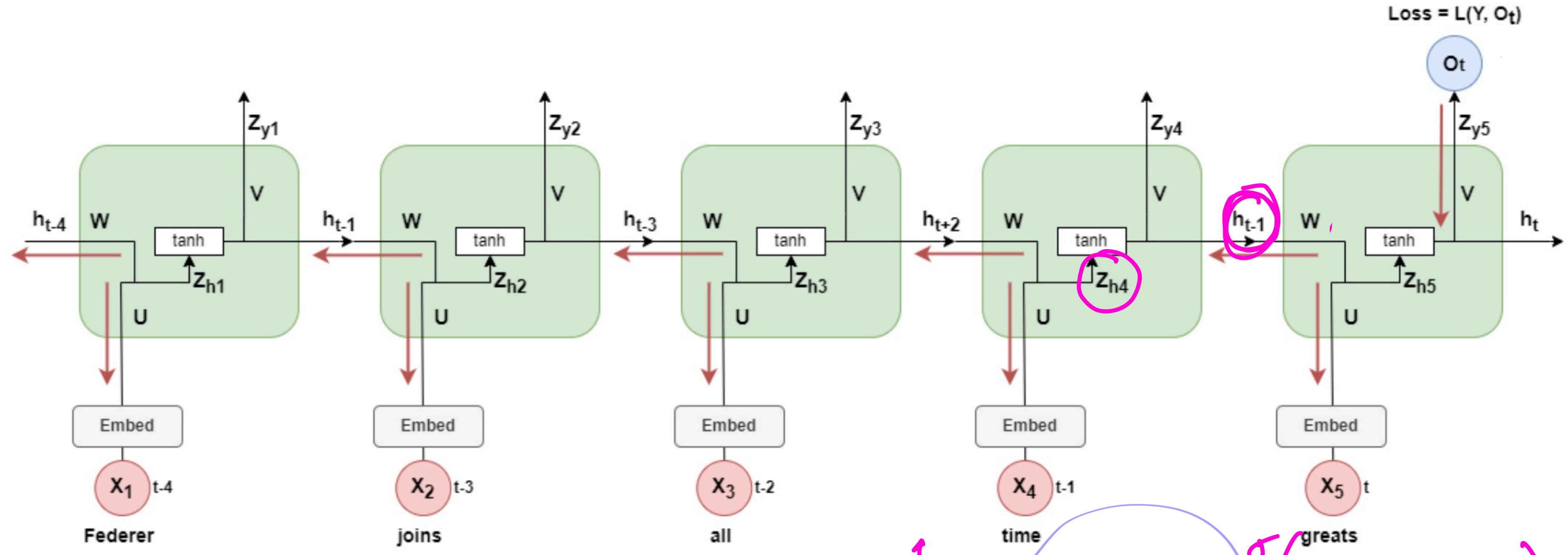
h_t → next hidden state.

x_t → Vector Representation of our input word at time t.

① OHE word.

② Fixed Word2Vec.

③ Trained Word 2 Vec.



$Z_{h1} = (X_1 * U) + (h_{t-5} * W) + b_h$
$h_1 = \sigma(Z_{h1})$, where σ = tanh
$Z_{y1} = (V * h_{t-4}) + b_y$

$Z_{h2} = (X_2 * U) + (h_{t-4} * W) + b_h$
$h_2 = \sigma(Z_{h2})$, where σ = tanh
$Z_{y2} = (V * h_{t-3}) + b_y$

$Z_{h3} = (X_3 * U) + (h_{t-3} * W) + b_h$
$h_3 = \sigma(Z_{h3})$, where σ = tanh
$Z_{y3} = (V * h_{t-2}) + b_y$

$Z_{h4} = (X_4 * U) + (h_{t-2} * W) + b_h$
$h_4 = \sigma(Z_{h4})$, where σ = tanh
$Z_{y4} = (V * h_{t-1}) + b_y$

$\sigma\left(Z_{h5} = (X_5 * U) + (h_{t-1} * W) + b_h\right)$
$h_5 = \sigma(Z_{h5})$, where σ = tanh
$Z_{y5} = (V * h_t) + b_y$
$O_{t+4} = \sigma(Z_{y5})$, where σ = Softmax

$$\nabla_{O_t} L \qquad \underline{\nabla_W L} = \underline{\nabla_{O_t} L} \times \underline{\nabla_{Z_{h5}} O_t} \times \underline{\nabla_W Z_{h5}}$$

(2nd time)

$$\nabla_W L = \nabla_{Z_{h4}} L \times \nabla_W Z_{h4}$$

$$Z_{h4} = (X_4 * U) + \sigma\left((X_5 * U) + h_{t-1} * W + b_h\right)(* W) + b_h$$

$k = 10.$