**CS224n: NLP with Deep Learning** **Winter 2019**

# Lecture 4

*Course Coordinator: Prof. Chris Manning* *Scribes: Akash Gupta*

**Deriving gradients wrt words for window model**:

The gradient that arrives at and updates the word vector can simply be split up for each word vector.

$$\delta_{window} = \begin{bmatrix} \nabla_{x_{aardvark}} \\ \nabla_{x_{in}} \\ \nabla_{x_{Paris}} \\ \nabla_{x_{arc}} \\ \nabla_{x_{amazing}} \end{bmatrix} \epsilon \, \mathbb{R}^{5d}$$

This pushes word vectors around so that it is more helpful in determining named entities.

**A pitfall in the above approach**:

Q) *What happens when we update the word vectors?*
A) Words in the training set like "TV" and "telly" move around but words in the testing set "television" stay at the same place. This can be bad as words vectors can be closer to each other in high-dimensional space but move farther away while updating.

- Can use pre-trained word vectors (Glove) since they are trained on huge amounts of data. If you have 100 millions of words in data then you can start with random word vectors.
- NOTE: you can update ("fine-tune") if you have a small training set and don't train the word vectors else if you have a large set then train = update = fine-tune word vectors to the task.

**Automatic Differentiation**:

- Automatically compute the gradient from the symbolic expression that we gave as input.
- Modern DL frameworks do backpropagation for you but mainly leave layer/node writer to hand-calculate the local derivative.

**Non-linearities**:

$->$ Sigmoid and tanh are slow to compute whereas ReLU is fast, easy to compute. Different versions - Leaky ReLU, parametric ReLU..

$->$ Parameter initialization - Xavier initialization - Tries to limit the values of W by looking at the previous layer size ($n_{in}$) and next layer size ($n_{out}$) so that it lies in the "slope" region of non-linearity functions.

$$Var(W_i) = \frac{2}{n_{in} + n_{out}}$$

$->$ Learning rates - (1) Constant (Start with lr = 0.001). (2) Try decreasing lr ($lr = lr \exp(-kt)$) (3) Fancy methods- Cyclic LR (increases and decreases)