Introduction and word vectors

- YouTube link
- Language is much younger than vision

Meaning of words

- Meaning in computers represented using synomyms and synsets in WordNet
- WordNet problems
  - hard to keep up to date
  - missing nuance (proficient is a synoynm for good)
  - no way to compute similarity

Representing words

- **traditional NLP**: words are discrete symbols (one-hot vectors)
- similarity with one-hot vectors is hard to measure (WordNet is incomplete, word-similarity tables don't scale –> Google did this in 2005)
- **modern NLP**: *A word's meaning is given by the words that frequently appear close-by*
- distributional semantics: words are represented by their context

Word2vec

- Word2vec is a framework for learning word vectors
- Idea: go through text and maximize similarity between words that appear in a context window
- likelihood: for each position, predict context words within a window of fixed size, given center word:

$$L(\theta) = \prod_{t=1}^{T} \prod_{-m \leq j \leq m} P(w_{t+j}|w_t;\theta)$$

- objective function is the negative log likelihood:

$$J(\theta) = -\frac{1}{T}\log L(\theta) = -\frac{1}{T}\sum_{t=1}^{T} \sum_{-m \leq j \leq m} \log P(w_{t+j}|w_t;\theta)$$

- minimizing the objective function is maximizing the likelihood
- how to calculate $P(w_{t+j}|w_t;\theta)$?
- v_w – vector word, u_w – context word
- dot product compares similarity between o (context) and c (center)

- normalize across entire vocabulary
- exponent makes the numbers way bigger (otherwise the distribution would be very flat)

$$P(o|c) = \frac{\exp(u_o^T v_c)}{\sum_{winV} \exp(u_w^T v_c)}$$

- to optimize, ones needs to minimize the function and compute all vector gradients (each word has two vectors)
- gradients are computed as partial derivatives of each softmax element

$$\frac{\partial}{\partial v_c} u_o^T v_c = u_o$$

$$\frac{\partial}{\partial v_c} \log \sum_{w=1}^{v} \exp(u_o^T v_c) = \frac{1}{\sum_{w=1}^{v} \exp(u_o v_c)} \sum_{x=1}^{w} \frac{\partial}{\partial v_c} exp(u_x^T v_c)$$

$$\frac{\partial}{\partial v_c} \log p(o|c) = u_o - \frac{\sum_{x=1}^{v} \exp(u_x^T v_c) u_x}{\sum_{w=1}^{v} \exp(u_w v_c} = u_o - \sum_{x=1}^{v} p(x|c) u_x$$

- we end up with the difference between the actual context word minus the expected context word