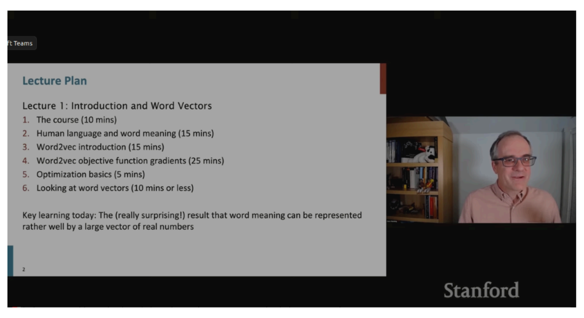
Lecture Notes

Monday, January 22, 2024 7:38 AM

Lecture Outline:



Intro to NLP

Human Languages:

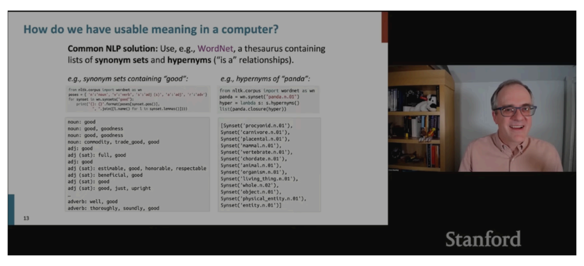
- Languages are means of sharing ideas, facts, intents etc.
- Languages have evolved/ use of languages have evolved to be useful for communication.
- Children vs Machines: Children learn a language very easily by interacting with the (multi-modal) world as compared machine. The Machines (advances In NLP) have no-where near the language-acquisition ability of Children.
- How to represent a language for better understanding of a Machine?? -> Deep learning provides the tools.
- The most important question to be answered by this course is -> How do we represent words? (to be better understood by the machines)

Uses of NLP:

- Machine Translation
- GPTs – Question answers, text generation and information retrieval
- Speech-to-text

Representing meaning of the word:

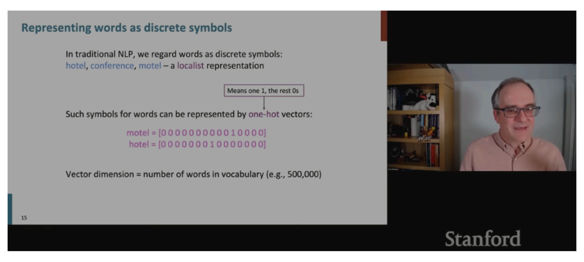
- What is Meaning? -> an idea behind a physical/imaginary concept
- Linguistic way of thinking about meaning?
  - A bond between signifier (I.e. the word, e.g. a chair) and signified (an idea behind the word, e.g. sitting)
- How to have usable meaning in computers? -> A dictionary

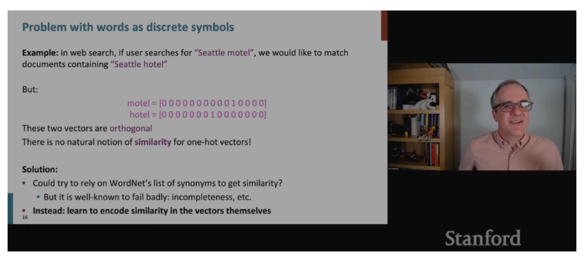


- Problem with WordNet:
  - Hard to always keep it up-to-date.
  - Details are missing: e.g. Good and Proficient have same meaning in some context but not always same.

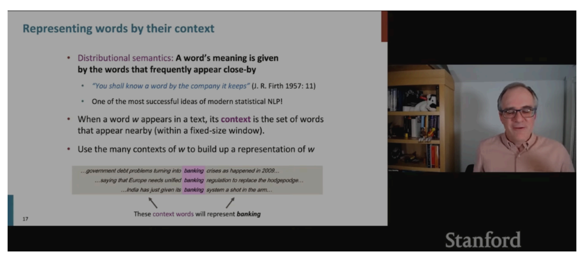
- Problems in traditional NLP ways:

  - Representing word as a discrete symbol and associated problems:
    - In short, the discrete vectors do not include context of the word and thus vectors of the similar words in context can be orthogonal





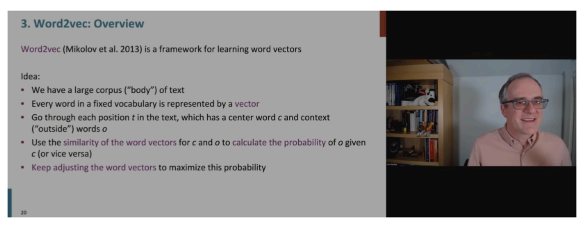
- Representing words by their context:

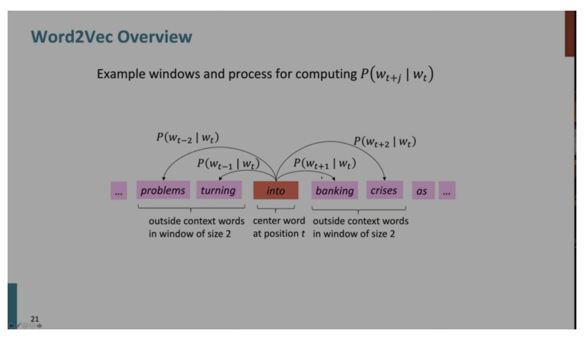


Word vectors: meaning of the word in context so that a word w1 has similar vector to word w2 if both of them are used in similar context.

Word2Vec:

- Overview:
  - Train a model that generates similar vector embeddings for words having similar in-context meaning.
  - During training, each word plays two roles. it becomes center word and also a context word.
  - Simple dot-product similarity is used to calculate similarity score between center-word and context-words vectors.
  - This similarity score is then converted to a probability distribution by using softmax



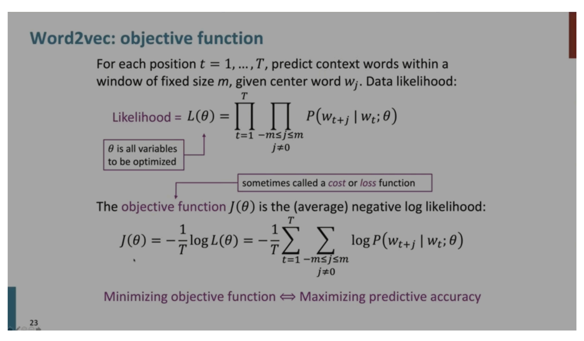


  - Likelihood Function:
    - Pick each word in the corpus as center word (Wj) and then predict the probability of context words (Wt+j) within a window of size m.

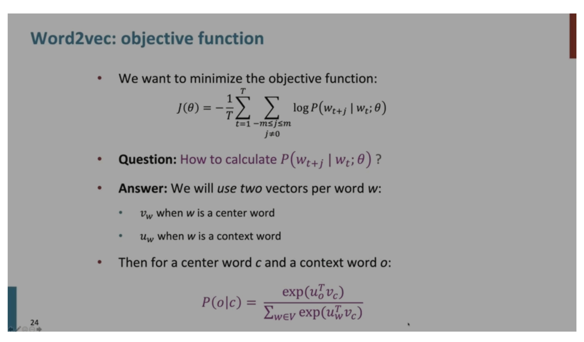
    - Repeat this process for all words and then take product.

  - Loss function:
    - Likelihood has to be maximized but the loss (the difference between model predicted likelihood and Ground truth likelihood) has to be minimized)
    - Hence take log of likelihood function and minimize the negative average of that log.

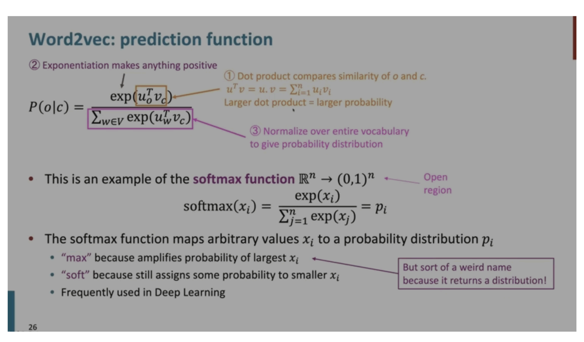
      Hint: $Log(a\,.\,b) = log(a) + log(b)$



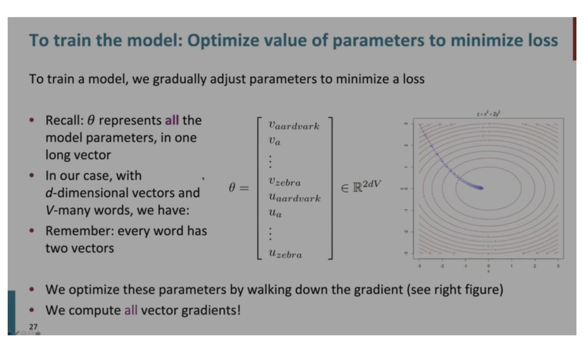
  - Calculating Probability of Context words given a center words
    - Why to assign two different vectors to a word when it is used a center word and a context word?



    - Use softmax to convert the word vectors into probabilities:



  - Training of Word2Vec model (just like any other model)



Partial derivative of Word2vec objective function.

$$J(\theta) = -\frac{1}{T} \sum_{t=1}^{T} \sum_{j \neq 0} \log P(w_{t+j} \mid w_t : \theta)$$

$$\because P(w_{t+j} \mid w_t) = \frac{\exp(u_o^T \cdot v_c)}{\sum_{w=1}^{V} \exp(u_w^T \cdot v_c)}$$

$$\frac{\partial}{\partial v_c} \log \frac{\exp(u_o^T \cdot v_c)}{\sum_{w=1}^{V} \exp(u_w^T \cdot v_c)} \longrightarrow \frac{\partial}{\partial}\left(\log \frac{a}{b}\right) = \log(a) - \log(b)$$

$$\underbrace{\frac{\partial}{\partial v_c} \log(\exp(u_o^T \cdot v_c))}_{\text{I}} - \underbrace{\log\left(\sum_{w=1}^{V} \exp(U_w^T \cdot v_c)\right)}_{\text{II}}$$

I $\log \exp(a) = a.$

$$\downarrow$$

$$\frac{\partial}{\partial v_c}(u_o^T \cdot v_c)$$

$$\downarrow$$

applying product Rule

$$\downarrow$$

$$u_o^T \cdot \frac{\partial}{\partial v_c} v_c + v_c \cdot \frac{\partial}{\partial v_c} u_o^T$$

$$\downarrow$$

$$u_o^T.$$

$$\frac{\partial}{\partial v_c} \underbrace{\log}_{f} \Big(\underbrace{\sum_{w=1}^{V} \exp(U_w^T \cdot v_c)}_{(z)} \Big) \Big\} \text{ chain rule } = \frac{\partial f(z)}{\partial v_c} = \frac{\partial f}{\partial z} \cdot \frac{\partial z}{\partial v_c}.$$

$$\frac{1}{\sum_{w=1}^{V} \exp(U_w^T \cdot v_c)} \cdot \frac{\partial}{\partial v_c} \sum_{x=1}^{V} \exp(u_x^T \cdot v_c) \quad \Big\} \ \frac{\partial}{\partial} \log(a) = \frac{1}{a} \leftarrow \text{ applying chain rule again.}$$

$$\cdot \sum_{x=1}^{V} \frac{\partial}{\partial v_c} \exp(u_x^T \cdot v_c)$$

$$\sum_{x=1}^{V} \exp(u_x^T \cdot v_c) \cdot \frac{\partial}{\partial v_c} u_x^T \cdot v_c$$

$$\sum_{x=1}^{V} \exp(u_x^T \cdot v_c) \cdot u_x$$

$$\frac{\sum_{x=1}^{V} \exp(u_x^T \cdot v_c) \cdot u_x}{\sum_{w=1}^{V} \exp(u_w^T \cdot v_c)}$$

Putting I & II together,

$$u_o - \frac{\sum_{x=1}^{V} \exp(u_x^T \cdot v_c) \cdot u_x}{\sum_{w=1}^{V} \exp(u_w^T \cdot v_c)}$$

$$u_o - \sum_{x=1}^{V} \boxed{\frac{\exp(u_x^T \cdot v_c)}{\sum_{w=1}^{V} \exp(U_w^T \cdot v_c)}} \cdot u_x \longrightarrow \text{softmax from input.}$$

$$u_o - \text{expected}$$

$\therefore$ observed − expected