

Precept 3: Word Embeddings

COS 484

Simon Park (slides borrowed from Tianyu Gao, Spring 2022)

2/14/2025

Today's Plan

1. Overview
2. Count-based methods
3. Predict-based methods
4. Evaluation
5. Matrix Calculus
6. Exercises

Overview

Overview - Word Embeddings

- Represent words as vectors
 - e.g., apple -> [0.1, 0.2, 0.5]
 - Encode semantic information
 - Useful for downstream NLP tasks

QUESTION: how can we get good word vectors

Overview - Distributional Hypothesis

- Words that occur in similar contexts have similar meaning
- EXAMPLE
 - A is the capital of ...
 - B is the capital of ...
- A, B should have similar meaning
- Word vectors for A, B should be “similar”

Overview - Different Approaches

- **Count-based** methods: PMI, PPMI, ...
 - Use statistics
- **Predict-based** methods: word2vec, GloVe, ...
 - Use ML

Count-based Methods

Word-word co-occurrence matrix W

- Choose a context window size (say 2)
- $W[t, c] = \# \text{ times } \mathbf{c} \text{ (context) appears in the context window of } \mathbf{t} \text{ (target)}$
- EXAMPLE
- When given sentence “I like cats and dogs,” increment the counts
- $W[\text{“I”}, \text{“like”}], W[\text{“I”}, \text{“cats”}]$
- $W[\text{“like”}, \text{“I”}], W[\text{“like”}, \text{“cats”}], W[\text{“like”}, \text{“and”}]$
- $W[\text{“cats”}, \text{“I”}], W[\text{“cats”}, \text{“like”}], W[\text{“cats”}, \text{“and”}], W[\text{“cats”}, \text{“dogs”}]$
- $W[\text{“and”}, \text{“like”}], W[\text{“and”}, \text{“cats”}], W[\text{“and”}, \text{“dogs”}]$
- $W[\text{“dogs”}, \text{“cats”}], W[\text{“dogs”}, \text{“and”}]$

Word-word co-occurrence matrix W

- $W[t, c] = \#$ times **c (context)** appears in the context window of **t (target)**

	I	like	...	dogs	
I	0	10		10	
like	10	0		40	
...					
dogs	10	40		0	

Word-word co-occurrence matrix W

- $W[t, c]$ = # times **c (context)** appears in the context window of **t (target)**
- **Quick poll: What is the size of W ?**

	I	like	...	dogs	
I	0	10		10	
like	10	0		40	
...					
dogs	10	40		0	

Word-word co-occurrence matrix W

- $W[t, c] = \#$ times **c (context)** appears in the context window of **t (target)**
- **Quick poll: What is the size of W ?** $|V| \times |V|$

	I	like	...	dogs	
I	0	10		10	
like	10	0		40	
...					
dogs	10	40		0	

Word-word co-occurrence matrix W

- $W[t, c] = \#$ times **c (context)** appears in the context window of **t (target)**

	I	like	...	dogs	
I	0	10		10	
like	10	0		40	
...					
dogs	10	40		0	

Word-word co-occurrence matrix W

- $W[t, c] = \#$ times **c (context)** appears in the context window of **t (target)**
- **Attempt 1: Word vector for word $t = W[t, :]$**

	I	like	...	dogs	
I	0	10		10	
like	10	0		40	
...					
dogs	10	40		0	

Attempt 1: Use raw frequency vector

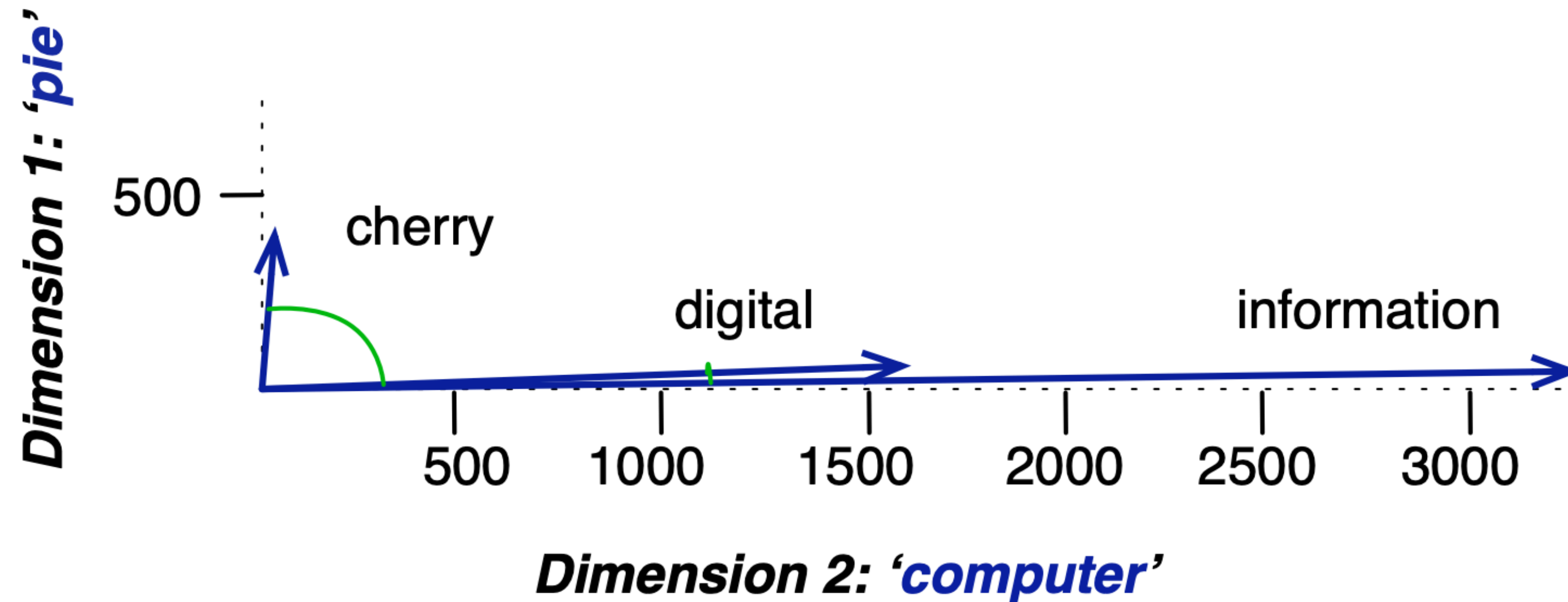
- Remember: Similar words should have **similar** word vectors
- First off, how do we measure the **similarity** between two vectors?

- Popular metric: cosine angle

- $$\cos(\mathbf{u}, \mathbf{v}) = \frac{\mathbf{u} \cdot \mathbf{v}}{\|\mathbf{u}\| \cdot \|\mathbf{v}\|}$$

Attempt 1: Use raw frequency vector

- Remember: Similar words should have **similar** word vectors



(Figure 6.8, J & M)

- When restricted to relevant sub-dimensions, kind of works out?

Attempt 1: Use raw frequency vector

- But some problems:
- Vectors are skewed by frequent words (e.g., “the”)
- ONE SOLUTION: check whether context word **c** occurs more frequently in the context of target word **t** than in other places
- Let’s go back to our word-word co-occurrence matrix

Word-word co-occurrence matrix W (ver. 2)

- $W[t, c] = \#$ times **c (context)** appears in the context window of **t (target)**

	I	like	...	dogs	
I	0	10		10	
like	10	0		40	
...					
dogs	10	40		0	

Word-word co-occurrence matrix W (ver. 2)

- $W[t, c] = \#$ times **c (context)** appears in the context window of **t (target)**
- First take the row / col sum

	I	like	...	dogs	count(t)
I	0	10		10	20
like	10	0		40	50
...					
dogs	10	40		0	50
count(c)	20	50		50	120

Word-word co-occurrence matrix W (ver. 2)

- $W[t, c] = \#$ times **c (context)** appears in the context window of **t (target)**
- Divide by the total sum

	I	like	...	dogs	count(t)
I	0	10		10	20
like	10	0		40	50
...					
dogs	10	40		0	50
count(c)	20	50		50	120

Word-word co-occurrence matrix W (ver. 2)

- $W[t, c]$ = fraction of times **c (context)** appears in the context of **t (target)**
- Divide by the total sum

	I	like	...	dogs	count(t)
I	0	1/12		1/12	1/6
like	1/12	0		1/3	5/12
...					
dogs	1/12	1/3		0	5/12
count(c)	1/6	5/12		5/12	1

Word-word co-occurrence matrix W (ver. 2)

- $W[t, c]$ = fraction of times **c (context)** appears in the context of **t (target)**
- $P["I"] = 1/6$ (probability that "I" appears)

	I	like	...	dogs	count(t)
I	0	1/12		1/12	1/6
like	1/12	0		1/3	5/12
...					
dogs	1/12	1/3		0	5/12
count(c)	1/6	5/12		5/12	1

Word-word co-occurrence matrix W (ver. 2)

- $W[t, c]$ = fraction of times **c (context)** appears in the context of **t (target)**
- **$P[\text{“like”}] = 5/12$** (probability that “like” appears)

	I	like	...	dogs	count(t)
I	0	1/12		1/12	1/6
like	1/12	0		1/3	5/12
...					
dogs	1/12	1/3		0	5/12
count(c)	1/6	5/12		5/12	1

Word-word co-occurrence matrix W (ver. 2)

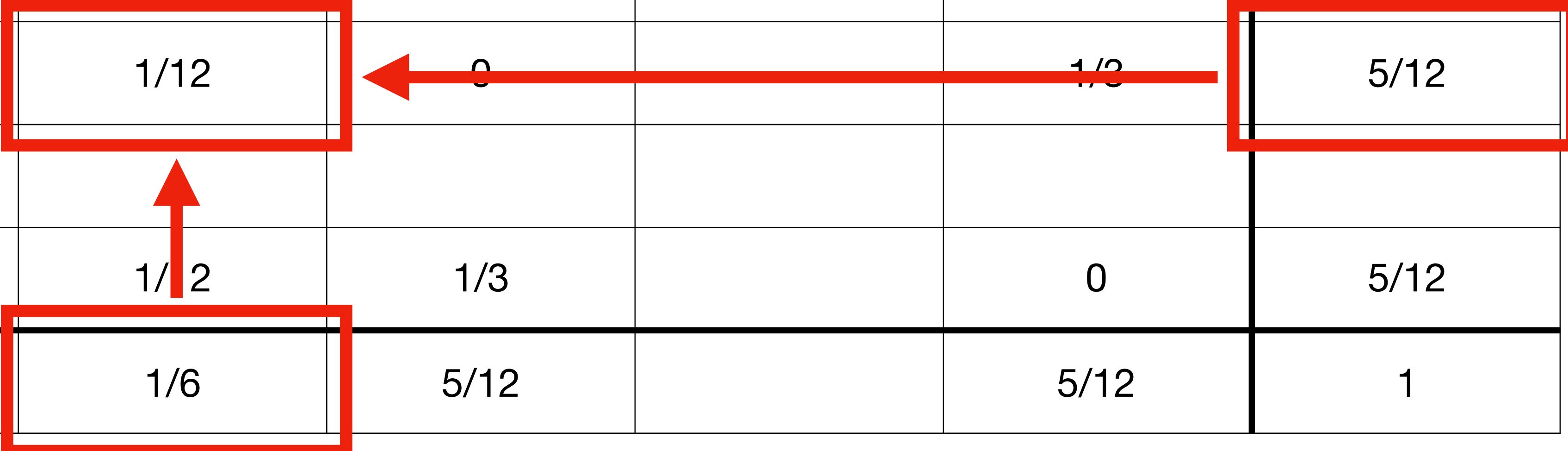
- $W[t, c]$ = fraction of times **c (context)** appears in the context of **t (target)**
- $P[\text{“like”, “I”}] = 1/12$ (probability that “I” appears in the context of “like”)

	I	like	...	dogs	count(t)
I	0	1/12		1/12	1/6
like	1/12	0		1/3	5/12
...					
dogs	1/12	1/3		0	5/12
count(c)	1/6	5/12		5/12	1

Word-word co-occurrence matrix W (ver. 2)

- If independent
- $P[\text{"like"}, \text{"I"}] = P[\text{"like"}] * P[\text{"I"}]$

	I	like	...	dogs	count(t)
I	0	1/12		1/12	1/6
like	1/12	0		1/3	5/12
...					
dogs	1/2	1/3		0	5/12
count(c)	1/6	5/12		5/12	1



Word-word co-occurrence matrix W (ver. 2)

- If related
- $P[\text{"like"}, \text{"I"}] > P[\text{"like"}] * P[\text{"I"}]$

	I	like	...	dogs	count(t)
I	0	1/12		1/12	1/6
like	1/12	0		1/3	5/12
...					
dogs	1/12	1/3		0	5/12
count(c)	1/6	5/12		5/12	1

Word-word co-occurrence matrix W (ver. 2)

- If unrelated
- $P[\text{"like"}, \text{"I"}] < P[\text{"like"}] * P[\text{"I"}]$

	I	like	...	dogs	count(t)
I	0	1/12		1/12	1/6
like	1/12	0		1/3	5/12
...					
dogs	1/12	1/3		0	5/12
count(c)	1/6	5/12		5/12	1

Word-word co-occurrence matrix W (ver. 2)

- Can we do this comparison more efficiently?

	I	like	...	dogs	count(t)
I	0	1/12		1/12	1/6
like	1/12	0		1/3	5/12
...					
dogs	1/12	1/3		0	5/12
count(c)	1/6	5/12		5/12	1

Word-word co-occurrence matrix W (ver. 2)

$$p(t, c) > p(t) \cdot p(c)$$

$$\Leftrightarrow \log p(t, c) > \log(p(t) \cdot p(c))$$

$$\Leftrightarrow \log p(t, c) - \log(p(t) \cdot p(c)) > 0$$

$$\Leftrightarrow \log \frac{p(t, c)}{p(t) \cdot p(c)} > 0$$

So define $PMI[t, c] = \log \frac{p(t, c)}{p(t)p(c)}$

PMI matrix (Pointwise Mutual Information)

- $PMI[t, c]$ = “excess possibility” **c (context)** appears in the context of **t (target)**
- **Attempt 2: Word vector for word t = $PMI[t, :]$**

	I	like	...	dogs	
I	-inf	0.18		0.18	
like	0.18	-inf		0.65	
...					
dogs	0.18	0.65		-inf	

Attempt 2: Use PMI Vector

- Main Problem: negative values of PMI are unreliable unless corpus is large
- ONE SOLUTION: clip all values at 0
- We only care about how much “relevant pairs” are relevant to each other and discard any information about “irrelevant pairs”
- Define $PPMI[t, c] = \max \left(\log \frac{p(t, c)}{p(t)p(c)}, 0 \right)$

PPMI matrix (Positive PMI)

- $PPMI[t, c]$ = “excess possibility” **c (context)** appears in the context of **t (target)**
- **Attempt 3: Word vector for word t = $PPMI[t, :]$**

	I	like	...	dogs	
I	0	0.18		0.18	
like	0.18	0		0.65	
...					
dogs	0.18	0.65		0	

PPMI matrix (Positive PMI)

- $PPMI[t, c]$ = “excess possibility” **c (context)** appears in the context of **t (target)**
- **Attempt 3: Word vector for word t = $PPMI[t, :]$**
- **Quick poll (recap): What is the size of PPMI matrix?**

	I	like	...	dogs	
I	0	0.18		0.18	
like	0.18	0		0.65	
...					
dogs	0.18	0.65		0	

PPMI matrix (Positive PMI)

- $PPMI[t, c]$ = “excess possibility” **c (context)** appears in the context of **t (target)**
- **Attempt 3: Word vector for word t = $PPMI[t, :]$**
- **Quick poll (recap): What is the size of PPMI matrix?** $|V| \times |V|$

	I	like	...	dogs	
I	0	0.18		0.18	
like	0.18	0		0.65	
...					
dogs	0.18	0.65		0	

Tackling Sparsity

- PPMI vectors are sparse
- Too many 0's
- Wasting dimensions
- ONE SOLUTION: Singular Value Decomposition (SVD)

$$PPMI = U\Sigma V^T$$

$$U, V : |V| \times k \text{ matrix}$$

$$\Sigma : k \times k \text{ matrix}$$

- **Attempt 4:**

Word vector for target word **t**: $U[t, :]$

Word vector for context word **c**: $V[c, :]$

Predict-based Methods

Overview

- Learn an ML model
- Input: **corpus**, dictionary **V**, and desired dimension **d**
- Output: learned model parameters
 - **embedding vector** of dimension **d** for each word

Word2vec / skip-gram

- Learn an ML model
- Input: corpus, dictionary V , and desired dimension d
- Output: learned model parameters
 - **two embedding vectors** of dimension d for each word
 - **u** when the word is a target word
 - **v** when the word is a context word

Word2vec / skip-gram

- Learn an ML model
- Input: corpus, dictionary V , and desired dimension d
- Output: learned model parameters
 - **two embedding vectors** of dimension d for each word
 - \mathbf{u} when the word is a target word
 - \mathbf{v} when the word is a context word
 - **Quick poll: How many model parameters total?**

Word2vec / skip-gram

- Learn an ML model
- Input: corpus, dictionary V , and desired dimension d
- Output: learned model parameters
 - **two embedding vectors** of dimension d for each word
 - \mathbf{u} when the word is a target word
 - \mathbf{v} when the word is a context word
 - **Quick poll: How many model parameters total?** $2d |V|$

Word2vec / skip-gram: Learning Objective Part 1

- Input: target word \mathbf{t} , context word \mathbf{c}
- Current model parameters: \mathbf{u} , \mathbf{v} for each word in dictionary
- Model's behavior:
 - Predict $\mathbb{P}[\mathbf{c} \mid \mathbf{t}]$
 - Probability that \mathbf{c} (out of all possible words) is in context of \mathbf{t}
 - How? Compute logits $\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}'}$ for all possible context words
 - Normalize with softmax function

- $$\mathbb{P}[\mathbf{c} \mid \mathbf{t}] = \frac{\exp(\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}})}{\sum_{\mathbf{c}'} \exp(\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}'})}$$

Word2vec / skip-gram: Learning Objective Part 1

- Input: target word \mathbf{t} , context word \mathbf{c}
- Current model parameters: \mathbf{u} , \mathbf{v} for each word in dictionary
- Model's behavior:

- $$\mathbb{P}[\mathbf{c} | \mathbf{t}] = \frac{\exp(\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}})}{\sum_{\mathbf{c}'} \exp(\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}'})}$$

- Loss = cross-entropy / negative log likelihood
- $L_{\mathbf{t},\mathbf{c}} = -\log \mathbb{P}[\mathbf{c} | \mathbf{t}]$

Word2vec / skip-gram: Learning Objective Part 2

- Input: a sequence of words w_1, w_2, \dots, w_T
- Model's behavior:
 - Choose a context window size m
 - Choose a word w_t and consider it a target word
 - For each word $w_{t-m}, \dots, w_{t-1}, w_{t+1}, \dots, w_{t+m}$ consider it a context word
 - Compute loss $L_{w_t, w_{t+j}}$ for all $-m \leq j \leq m, j \neq 0$
 - Sum it all up
$$L_t = \sum_{-m \leq j \leq m, j \neq 0} L_{w_t, w_{t+j}}$$
- Model's loss for predicting context words of a particular word w_t

Word2vec / skip-gram: Learning Objective Part 3

- Input: a sequence of words w_1, w_2, \dots, w_T

- Model's behavior:

- $$L_t = \sum_{-m \leq j \leq m, j \neq 0} L_{w_t, w_{t+j}}$$

- Model's loss for predicting context words of a particular word w_t

- Take the average to get the final loss
$$L = \frac{1}{T} \sum_{t=1}^T L_t$$

Word2vec / skip-gram: Learning Objective Part 3

- Input: a sequence of words w_1, w_2, \dots, w_T

- Model's behavior:

- $$L = -\frac{1}{T} \sum_{t=1}^T \sum_{-m \leq j \leq m, j \neq 0} \log \frac{\exp(\mathbf{u}_{w_t} \cdot \mathbf{v}_{w_{t+j}})}{\sum_{\mathbf{c} \in V} \exp(\mathbf{u}_{w_t} \cdot \mathbf{v}_{\mathbf{c}})}$$

- Note $\mathbf{c} \in V$ takes the sum over all possible words in the dictionary
- Not just the words that appear in the corpus / sequence of words

Word2vec / skip-gram: How to Train

- Input: a sequence of words w_1, w_2, \dots, w_T

- Compute loss:

$$L = -\frac{1}{T} \sum_{t=1}^T \sum_{-m \leq j \leq m, j \neq 0} \log \frac{\exp(\mathbf{u}_{w_t} \cdot \mathbf{v}_{w_{t+j}})}{\sum_{\mathbf{c} \in V} \exp(\mathbf{u}_{w_t} \cdot \mathbf{v}_{\mathbf{c}})}$$

- Compute gradient of the loss with respect to all $\mathbf{u}, \mathbf{v} : \frac{\partial L}{\partial \mathbf{u}}, \frac{\partial L}{\partial \mathbf{v}}$
- Update model parameters via Gradient Descent
- Problem? For every pair (\mathbf{t}, \mathbf{c}) , you need to update $2d \mid V \mid$ parameters
- ONE SOLUTION: instead of $\mathbf{c} \in V$, only sample K (5-20) alternatives

Word2vec / skip-gram: Learning Objective Part 1 (recap)

- Input: target word \mathbf{t} , context word \mathbf{c}
- Current model parameters: \mathbf{u} , \mathbf{v} for each word in dictionary
- Model's behavior:
 - Predict $\mathbb{P}[\mathbf{c} \mid \mathbf{t}]$
 - Probability that \mathbf{c} (out of all possible words) is in context of \mathbf{t}
 - How? Compute logits $\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}'}$ for all possible context words
 - Normalize with softmax function

- $$\mathbb{P}[\mathbf{c} \mid \mathbf{t}] = \frac{\exp(\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}})}{\sum_{\mathbf{c}'} \exp(\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}'})}$$

Word2vec / skip-gram: Negative Sampling Part 1

- Input: target word \mathbf{t} , context word \mathbf{c} , **alternative context words** $\mathbf{c}_1, \dots, \mathbf{c}_K$
- Current model parameters: \mathbf{u} , \mathbf{v} for each word in dictionary
- Model's behavior:

- Predict $\mathbb{P}[+ | \mathbf{t}, \mathbf{c}] \cdot \prod_{i=1}^K \mathbb{P}[- | \mathbf{t}, \mathbf{c}_i]$

- Probability that \mathbf{c} is in context of \mathbf{t} **AND** $\mathbf{c}_1, \dots, \mathbf{c}_K$ are not in context of \mathbf{t}
- How? Consider these **independent** binary logistic regression
- $\mathbb{P}[+ | \mathbf{t}, \mathbf{c}] = \sigma(\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}})$
- $\mathbb{P}[- | \mathbf{t}, \mathbf{c}_i] = \sigma(-\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}_i})$

Word2vec / skip-gram: Negative Sampling Part 1

- Input: target word \mathbf{t} , context word \mathbf{c} , **alternative context words** $\mathbf{c}_1, \dots, \mathbf{c}_K$
- Current model parameters: \mathbf{u} , \mathbf{v} for each word in dictionary
- Model's behavior:

- $\mathbb{P}[+ | \mathbf{t}, \mathbf{c}] \cdot \prod_{i=1}^K \mathbb{P}[- | \mathbf{t}, \mathbf{c}_i] = \sigma(\mathbf{u}_t \cdot \mathbf{v}_c) \cdot \prod_{i=1}^K \sigma(-\mathbf{u}_t \cdot \mathbf{v}_{c_i})$

- Loss = cross-entropy / negative log likelihood

- $L_{t,c} = -\log \sigma(\mathbf{u}_t \cdot \mathbf{v}_c) - \sum_{i=1}^K \log \sigma(-\mathbf{u}_t \cdot \mathbf{v}_{c_i})$

- Recall that $\mathbf{c}_1, \dots, \mathbf{c}_K$ randomly sampled

Word2vec / skip-gram: Negative Sampling Part 1

- Input: target word \mathbf{t} , context word \mathbf{c} , **alternative context words** $\mathbf{c}_1, \dots, \mathbf{c}_K$
- Current model parameters: \mathbf{u}, \mathbf{v} for each word in dictionary
- Model's behavior:

- $\mathbb{P}[+ | \mathbf{t}, \mathbf{c}] \cdot \prod_{i=1}^K \mathbb{P}[- | \mathbf{t}, \mathbf{c}_i] = \sigma(\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}}) \cdot \prod_{i=1}^K \sigma(-\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}_i})$

- Loss = cross-entropy / negative log likelihood

- $$L_{\mathbf{t}, \mathbf{c}} = -\log \sigma(\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}}) - \sum_{i=1}^K \mathbb{E}_{\mathbf{c}_i \sim V} \log \sigma(-\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}_i})$$

- Recall that $\mathbf{c}_1, \dots, \mathbf{c}_K$ randomly sampled

Word2vec / skip-gram: Negative Sampling Part 1

- Input: target word \mathbf{t} , context word \mathbf{c} , **alternative context words** $\mathbf{c}_1, \dots, \mathbf{c}_K$
- Current model parameters: \mathbf{u}, \mathbf{v} for each word in dictionary
- Model's behavior:

- $\mathbb{P}[+ | \mathbf{t}, \mathbf{c}] \cdot \prod_{i=1}^K \mathbb{P}[- | \mathbf{t}, \mathbf{c}_i] = \sigma(\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}}) \cdot \prod_{i=1}^K \sigma(-\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}_i})$

- Loss = cross-entropy / negative log likelihood

- $$L_{\mathbf{t}, \mathbf{c}} = -\log \sigma(\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}}) - \sum_{i=1}^K \mathbb{E}_{\mathbf{c}_i \sim V} \log \sigma(-\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}_i})$$

- **Quick Poll:** How many parameters contribute to this loss value?

Word2vec / skip-gram: Negative Sampling Part 1

- Input: target word \mathbf{t} , context word \mathbf{c} , **alternative context words** $\mathbf{c}_1, \dots, \mathbf{c}_K$
- Current model parameters: \mathbf{u}, \mathbf{v} for each word in dictionary
- Model's behavior:

- $\mathbb{P}[+ | \mathbf{t}, \mathbf{c}] \cdot \prod_{i=1}^K \mathbb{P}[- | \mathbf{t}, \mathbf{c}_i] = \sigma(\mathbf{u}_t \cdot \mathbf{v}_c) \cdot \prod_{i=1}^K \sigma(-\mathbf{u}_t \cdot \mathbf{v}_{c_i})$

- Loss = cross-entropy / negative log likelihood

- $$L_{\mathbf{t}, \mathbf{c}} = -\log \sigma(\mathbf{u}_t \cdot \mathbf{v}_c) - \sum_{i=1}^K \mathbb{E}_{\mathbf{c}_i \sim V} \log \sigma(-\mathbf{u}_t \cdot \mathbf{v}_{c_i})$$

- **Quick Poll: How many parameters contribute to this loss value?**

- $(K + 2)d$ (d for each of $\mathbf{u}_t, \mathbf{v}_t, \mathbf{v}_{c_1}, \mathbf{v}_{c_2}, \dots, \mathbf{v}_{c_K}$)

Word2vec / skip-gram: Train with Negative Sampling

- Input: a sequence of words w_1, w_2, \dots, w_T

- Compute loss:

$$L = \frac{1}{T} \sum_{t=1}^T \sum_{-m \leq j \leq m, j \neq 0} L_{w_t, w_{t+j}}$$

- Compute gradient of the loss with respect to all $\mathbf{u}, \mathbf{v} : \frac{\partial L}{\partial \mathbf{u}}, \frac{\partial L}{\partial \mathbf{v}}$
- Update model parameters via Gradient Descent

Evaluation

Overview

- Recall earlier QUESTION: **how can we get good word vectors**
- How do we know if we got **good** word vectors?
- Intrinsic Evaluation
 - evaluate word vectors directly
 - “similarity” based tasks
- Extrinsic Evaluation:
 - evaluate ML model built on top of the word vectors
 - other tasks

Matrix Calculus

Matrix Calculus / Vectorized Gradients

- Go through this note:
- <http://web.stanford.edu/class/cs224n/readings/gradient-notes.pdf>
- Make sure that you can understand all the cases in Section 2, 3

Basic Definition

$$f: \mathbb{R}^n \longrightarrow \mathbb{R}^m$$

$$f(\mathbf{x}) = (f_1(\mathbf{x}), \cdots, f_m(\mathbf{x}))$$

$$\frac{\partial f}{\partial \mathbf{x}} = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_m}{\partial x_1} & \cdots & \frac{\partial f_m}{\partial x_n} \end{bmatrix}$$

Gradient w.r.t vector - matrix * vector

Can directly take gradients of a vector with respect to a vector

$$\mathbf{x} \in \mathbb{R}^n \quad \mathbf{z} \in \mathbb{R}^m \quad \mathbf{W} \in \mathbb{R}^{m \times n}$$

$$\mathbf{z} = \mathbf{W}\mathbf{x}$$

$$\frac{\partial \mathbf{z}}{\partial \mathbf{x}} = \mathbf{W}$$

Gradient w.r.t vector - vector * matrix

Can directly take gradients of a vector with respect to a vector

$$\mathbf{x} \in \mathbb{R}^n \quad \mathbf{z} \in \mathbb{R}^m \quad \mathbf{W} \in \mathbb{R}^{n \times m}$$

$$\mathbf{z} = \mathbf{x}\mathbf{W}$$

$$\frac{\partial \mathbf{z}}{\partial \mathbf{x}} = \mathbf{W}^T$$

Gradient w.r.t vector - elementwise operation

Can directly take gradients of a vector with respect to a vector

$$\mathbf{x} \in \mathbb{R}^n \quad \mathbf{z} \in \mathbb{R}^n \quad \mathbf{f} \in \mathbb{R}^n \longrightarrow \mathbb{R}^n$$

$$\mathbf{z} = f(\mathbf{x}) = (f(x_1), \dots, f(x_n))$$

$$\frac{\partial \mathbf{z}}{\partial \mathbf{x}} = \text{diag}(f'(\mathbf{x})) = \begin{bmatrix} f'(x_1) & 0 & \dots & 0 \\ 0 & f'(x_2) & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & f'(x_n) \end{bmatrix}$$

Gradient w.r.t matrix - matrix * vector

Cannot directly take gradients of a vector with respect to a matrix

Need to take gradient of the final loss value (scalar) with respect to matrix

$$\mathbf{x} \in \mathbb{R}^n \quad \mathbf{z} \in \mathbb{R}^m \quad \mathbf{W} \in \mathbb{R}^{m \times n} \quad L \in \mathbb{R}$$

$$\mathbf{z} = \mathbf{W}\mathbf{x} \quad L = f(\mathbf{z})$$

$$\frac{\partial L}{\partial \mathbf{W}} = \left(\frac{\partial L}{\partial \mathbf{z}} \right)^T \mathbf{x}^T$$

Gradient w.r.t matrix - vector * matrix

Cannot directly take gradients of a vector with respect to a matrix

Need to take gradient of the final loss value (scalar) with respect to matrix

$$\mathbf{x} \in \mathbb{R}^n \quad \mathbf{z} \in \mathbb{R}^m \quad \mathbf{W} \in \mathbb{R}^{n \times m} \quad L \in \mathbb{R}$$

$$\mathbf{z} = \mathbf{x}\mathbf{W} \quad L = f(\mathbf{z})$$

$$\frac{\partial L}{\partial \mathbf{W}} = \mathbf{x}^T \left(\frac{\partial L}{\partial \mathbf{z}} \right)$$

Exercises

Q1: Pointwise Mutual Information (PMI)

Recall the formula for PMI between words w_1, w_2

$$PMI(w_1, w_2) = \log \frac{P(w_1, w_2)}{P(w_1)P(w_2)}$$

- (a) What does it mean in terms of $P(w_1, w_2)$ and $P(w_1)P(w_2)$ when PMI is negative? What are some examples of when this might happen?
- (b) Remember that **positive PMI (PPMI)** floors the PMI at 0. Why does this make sense for the scenario from (a)?

Q1: Pointwise Mutual Information (PMI)

$$PMI(w_1, w_2) = \log \frac{P(w_1, w_2)}{P(w_1)P(w_2)}$$

(a) What does it mean in terms of $P(w_1, w_2)$ and $P(w_1)P(w_2)$ when PMI is negative? What are some examples of when this might happen?

In this case, the joint probability is less than the product of the marginal probabilities. This means the words occur **less often than if they were independent**

Consider $P(w_1) = P(w_2) = 10^{-6}$ which are very infrequent words. It is unlikely to observe a single case of w_1, w_2 occurring together (in a reasonably sized corpus) and $P(w_1, w_2) = 0$ leading to $PMI = -\infty$

Q1: Pointwise Mutual Information (PMI)

$$PMI(w_1, w_2) = \log \frac{P(w_1, w_2)}{P(w_1)P(w_2)}$$

(b) Remember that **positive PMI (PPMI)** floors the PMI at 0. Why does this make sense for the scenario from (a)?

In the case where w_1, w_2 are very infrequent words, the negative PMI values were not meaningful. Flooring at 0 might mitigate some noise.

Q2: Gradients for Skip-gram with Negative Sampling

Recall: loss for target \mathbf{t} , context \mathbf{c} , **alternative context** $\mathbf{c}_1, \dots, \mathbf{c}_K$

$$L_{\mathbf{t},\mathbf{c}} = -\log \sigma(\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}}) - \sum_{i=1}^K \log \sigma(-\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}_i})$$

(a) $\frac{\partial L_{\mathbf{t},\mathbf{c}}}{\partial \mathbf{u}_{\mathbf{t}}} = ?$

(b) $\frac{\partial L_{\mathbf{t},\mathbf{c}}}{\partial \mathbf{v}_{\mathbf{c}}} = ?$

(c) $\frac{\partial L_{\mathbf{t},\mathbf{c}}}{\partial \mathbf{v}_{\mathbf{c}_j}} = ?$

A particular index j (not the same index i in the sum)

Q2: (a)

$$L_{\mathbf{t},\mathbf{c}} = -\log \sigma(\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}}) - \sum_{i=1}^K \log \sigma(-\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}_i})$$

$$\frac{\partial L_{\mathbf{t},\mathbf{c}}}{\partial \mathbf{u}_{\mathbf{t}}} =$$

Q2: (a)

$$L_{\mathbf{t},\mathbf{c}} = -\log \sigma(\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}}) - \sum_{i=1}^K \log \sigma(-\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}_i})$$

$$\frac{\partial L_{\mathbf{t},\mathbf{c}}}{\partial \mathbf{u}_{\mathbf{t}}} = -\frac{\frac{\partial \sigma(\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}})}{\partial \mathbf{u}_{\mathbf{t}}}}{\sigma(\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}})} - \sum_{i=1}^K \frac{\frac{\partial \sigma(-\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}_i})}{\partial \mathbf{u}_{\mathbf{t}}}}{\sigma(-\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}_i})}$$

1. $d \log(x) / dx = 1/x$
2. chain rule

Q2: (a)

$$L_{t,c} = -\log \sigma(\mathbf{u}_t \cdot \mathbf{v}_c) - \sum_{i=1}^K \log \sigma(-\mathbf{u}_t \cdot \mathbf{v}_{c_i})$$

$$\frac{\partial L_{t,c}}{\partial \mathbf{u}_t} = -\frac{\frac{\partial \sigma(\mathbf{u}_t \cdot \mathbf{v}_c)}{\partial \mathbf{u}_t}}{\sigma(\mathbf{u}_t \cdot \mathbf{v}_c)} - \sum_{i=1}^K \frac{\frac{\partial \sigma(-\mathbf{u}_t \cdot \mathbf{v}_{c_i})}{\partial \mathbf{u}_t}}{\sigma(-\mathbf{u}_t \cdot \mathbf{v}_{c_i})}$$

1. $\frac{d \text{sigmoid}(x)}{dx} = \text{sig}(x) * (1 - \text{sig}(x))$
2. chain rule

$$= -\frac{\sigma(\mathbf{u}_t \cdot \mathbf{v}_c)(1 - \sigma(\mathbf{u}_t \cdot \mathbf{v}_c)) \frac{\partial(\mathbf{u}_t \cdot \mathbf{v}_c)}{\partial \mathbf{u}_t}}{\sigma(\mathbf{u}_t \cdot \mathbf{v}_c)} - \sum_{i=1}^K \frac{\sigma(-\mathbf{u}_t \cdot \mathbf{v}_{c_i})(1 - \sigma(-\mathbf{u}_t \cdot \mathbf{v}_{c_i})) \frac{\partial(-\mathbf{u}_t \cdot \mathbf{v}_{c_i})}{\partial \mathbf{u}_t}}{\sigma(-\mathbf{u}_t \cdot \mathbf{v}_{c_i})}$$

Q2: (a)

$$L_{t,c} = -\log \sigma(\mathbf{u}_t \cdot \mathbf{v}_c) - \sum_{i=1}^K \log \sigma(-\mathbf{u}_t \cdot \mathbf{v}_{c_i})$$

$$\begin{aligned} \frac{\partial L_{t,c}}{\partial \mathbf{u}_t} &= -\frac{\frac{\partial \sigma(\mathbf{u}_t \cdot \mathbf{v}_c)}{\partial \mathbf{u}_t}}{\sigma(\mathbf{u}_t \cdot \mathbf{v}_c)} - \sum_{i=1}^K \frac{\frac{\partial \sigma(-\mathbf{u}_t \cdot \mathbf{v}_{c_i})}{\partial \mathbf{u}_t}}{\sigma(-\mathbf{u}_t \cdot \mathbf{v}_{c_i})} \quad \text{d (x * y) / dx = y} \\ &= -\frac{\sigma(\mathbf{u}_t \cdot \mathbf{v}_c)(1 - \sigma(\mathbf{u}_t \cdot \mathbf{v}_c)) \frac{\partial(\mathbf{u}_t \cdot \mathbf{v}_c)}{\partial \mathbf{u}_t}}{\sigma(\mathbf{u}_t \cdot \mathbf{v}_c)} - \sum_{i=1}^K \frac{\sigma(-\mathbf{u}_t \cdot \mathbf{v}_{c_i})(1 - \sigma(-\mathbf{u}_t \cdot \mathbf{v}_{c_i})) \frac{\partial(-\mathbf{u}_t \cdot \mathbf{v}_{c_i})}{\partial \mathbf{u}_t}}{\sigma(-\mathbf{u}_t \cdot \mathbf{v}_{c_i})} \\ &= -\frac{\sigma(\mathbf{u}_t \cdot \mathbf{v}_c)(1 - \sigma(\mathbf{u}_t \cdot \mathbf{v}_c))\mathbf{v}_c}{\sigma(\mathbf{u}_t \cdot \mathbf{v}_c)} - \sum_{i=1}^K \frac{\sigma(-\mathbf{u}_t \cdot \mathbf{v}_{c_i})(1 - \sigma(-\mathbf{u}_t \cdot \mathbf{v}_{c_i}))(-\mathbf{v}_{c_i})}{\sigma(-\mathbf{u}_t \cdot \mathbf{v}_{c_i})} \end{aligned}$$

Q2: (a)

$$L_{\mathbf{t},\mathbf{c}} = -\log \sigma(\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}}) - \sum_{i=1}^K \log \sigma(-\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}_i})$$

$$\begin{aligned} \frac{\partial L_{\mathbf{t},\mathbf{c}}}{\partial \mathbf{u}_{\mathbf{t}}} &= -\frac{\frac{\partial \sigma(\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}})}{\partial \mathbf{u}_{\mathbf{t}}}}{\sigma(\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}})} - \sum_{i=1}^K \frac{\frac{\partial \sigma(-\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}_i})}{\partial \mathbf{u}_{\mathbf{t}}}}{\sigma(-\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}_i})} \\ &= -\frac{\sigma(\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}})(1 - \sigma(\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}})) \frac{\partial(\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}})}{\partial \mathbf{u}_{\mathbf{t}}}}{\sigma(\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}})} - \sum_{i=1}^K \frac{\sigma(-\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}_i})(1 - \sigma(-\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}_i})) \frac{\partial(-\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}_i})}{\partial \mathbf{u}_{\mathbf{t}}}}{\sigma(-\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}_i})} \\ &= -\frac{\cancel{\sigma(\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}})}(1 - \sigma(\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}}))\mathbf{v}_{\mathbf{c}}}{\cancel{\sigma(\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}})}} - \sum_{i=1}^K \frac{\cancel{\sigma(-\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}_i})}(1 - \sigma(-\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}_i}))(-\mathbf{v}_{\mathbf{c}_i})}{\cancel{\sigma(-\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}_i})}} \end{aligned}$$

Q2: (a)

$$L_{\mathbf{t},\mathbf{c}} = -\log \sigma(\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}}) - \sum_{i=1}^K \log \sigma(-\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}_i})$$

$$\begin{aligned} \frac{\partial L_{\mathbf{t},\mathbf{c}}}{\partial \mathbf{u}_{\mathbf{t}}} &= -\frac{\frac{\partial \sigma(\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}})}{\partial \mathbf{u}_{\mathbf{t}}}}{\sigma(\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}})} - \sum_{i=1}^K \frac{\frac{\partial \sigma(-\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}_i})}{\partial \mathbf{u}_{\mathbf{t}}}}{\sigma(-\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}_i})} \\ &= -\frac{\sigma(\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}})(1 - \sigma(\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}}))\frac{\partial(\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}})}{\partial \mathbf{u}_{\mathbf{t}}}}{\sigma(\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}})} - \sum_{i=1}^K \frac{\sigma(-\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}_i})(1 - \sigma(-\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}_i}))\frac{\partial(-\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}_i})}{\partial \mathbf{u}_{\mathbf{t}}}}{\sigma(-\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}_i})} \\ &= -(1 - \sigma(\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}}))\mathbf{v}_{\mathbf{c}} - \sum_{i=1}^K (1 - \sigma(-\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}_i}))(-\mathbf{v}_{\mathbf{c}_i}) \end{aligned}$$

Q2: (a)

$$L_{\mathbf{t},\mathbf{c}} = -\log \sigma(\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}}) - \sum_{i=1}^K \log \sigma(-\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}_i})$$

$$\begin{aligned} \frac{\partial L_{\mathbf{t},\mathbf{c}}}{\partial \mathbf{u}_{\mathbf{t}}} &= -\frac{\frac{\partial \sigma(\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}})}{\partial \mathbf{u}_{\mathbf{t}}}}{\sigma(\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}})} - \sum_{i=1}^K \frac{\frac{\partial \sigma(-\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}_i})}{\partial \mathbf{u}_{\mathbf{t}}}}{\sigma(-\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}_i})} \\ &= -\frac{\sigma(\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}})(1 - \sigma(\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}}))\frac{\partial(\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}})}{\partial \mathbf{u}_{\mathbf{t}}}}{\sigma(\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}})} - \sum_{i=1}^K \frac{\sigma(-\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}_i})(1 - \sigma(-\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}_i}))\frac{\partial(-\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}_i})}{\partial \mathbf{u}_{\mathbf{t}}}}{\sigma(-\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}_i})} \\ &= -(1 - \sigma(\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}}))\mathbf{v}_{\mathbf{c}} - \sum_{i=1}^K (1 - \sigma(-\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}_i}))(-\mathbf{v}_{\mathbf{c}_i}) \\ &= (\sigma(\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}}) - 1)\mathbf{v}_{\mathbf{c}} + \sum_{i=1}^K \sigma(\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}_i})\mathbf{v}_{\mathbf{c}_i} \quad \text{sigmoid}(-\mathbf{x}) = 1 - \text{sigmoid}(\mathbf{x}) \end{aligned}$$

Q2: (a) FINAL VERSION

$$\frac{\partial L_{\mathbf{t},\mathbf{c}}}{\partial \mathbf{u}_{\mathbf{t}}} = (\sigma(\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}}) - 1)\mathbf{v}_{\mathbf{c}} + \sum_{i=1}^K \sigma(\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}_i})\mathbf{v}_{\mathbf{c}_i}$$

Q2: (b)

$$L_{\mathbf{t},\mathbf{c}} = -\log \sigma(\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}}) - \sum_{i=1}^K \log \sigma(-\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}_i})$$

$$\frac{\partial L_{\mathbf{t},\mathbf{c}}}{\partial \mathbf{v}_{\mathbf{c}}} =$$

Q2: (b)

$$L_{\mathbf{t},\mathbf{c}} = -\log \sigma(\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}}) - \sum_{i=1}^K \log \sigma(-\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}_i})$$

$$\frac{\partial L_{\mathbf{t},\mathbf{c}}}{\partial \mathbf{v}_{\mathbf{c}}} = - \frac{\frac{\partial \sigma(\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}})}{\partial \mathbf{v}_{\mathbf{c}}}}{\sigma(\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}})}$$

Summands do not depend on $\mathbf{v}_{\mathbf{c}}$

Q2: (b)

$$L_{\mathbf{t},\mathbf{c}} = -\log \sigma(\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}}) - \sum_{i=1}^K \log \sigma(-\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}_i})$$

$$\frac{\partial L_{\mathbf{t},\mathbf{c}}}{\partial \mathbf{v}_{\mathbf{c}}} = - \frac{\frac{\partial \sigma(\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}})}{\partial \mathbf{v}_{\mathbf{c}}}}{\sigma(\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}})}$$

$$= -(1 - \sigma(\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}}))\mathbf{u}_{\mathbf{t}}$$

$$= (\sigma(\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}}) - 1)\mathbf{u}_{\mathbf{t}}$$

Same as before

Q2: (b) FINAL VERSION

$$\frac{\partial L_{\mathbf{t},\mathbf{c}}}{\partial \mathbf{v}_{\mathbf{c}}} = (\sigma(\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}}) - 1)\mathbf{u}_{\mathbf{t}}$$

Q2: (c)

$$L_{\mathbf{t},\mathbf{c}} = -\log \sigma(\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}}) - \sum_{i=1}^K \log \sigma(-\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}_i})$$

$$\frac{\partial L_{\mathbf{t},\mathbf{c}}}{\partial \mathbf{v}_{\mathbf{c}_j}} =$$

Q2: (c)

$$L_{\mathbf{t},\mathbf{c}} = -\log \sigma(\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}}) - \sum_{i=1}^K \log \sigma(-\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}_i})$$

$$\frac{\partial L_{\mathbf{t},\mathbf{c}}}{\partial \mathbf{v}_{\mathbf{c}_j}} = - \frac{\frac{\partial \sigma(-\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}_j})}{\partial \mathbf{v}_{\mathbf{c}_j}}}{\sigma(-\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}_j})}$$

All other summands do not depend on j

Q2: (c)

$$L_{\mathbf{t},\mathbf{c}} = -\log \sigma(\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}}) - \sum_{i=1}^K \log \sigma(-\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}_i})$$

$$\frac{\partial L_{\mathbf{t},\mathbf{c}}}{\partial \mathbf{v}_{\mathbf{c}_j}} = - \frac{\frac{\partial \sigma(-\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}_j})}{\partial \mathbf{v}_{\mathbf{c}_j}}}{\sigma(-\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}_j})}$$

$$= - (1 - \sigma(-\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}_j}))(-\mathbf{u}_{\mathbf{t}})$$

Same as before

$$= \sigma(\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}_j})\mathbf{u}_{\mathbf{t}}$$

Q2: (c) FINAL VERSION

$$\frac{\partial L_{t,c}}{\partial \mathbf{v}_{c_j}} = \sigma(\mathbf{u}_t \cdot \mathbf{v}_{c_j}) \mathbf{u}_t$$

Q2: FINAL VERSION

Recall: loss for target \mathbf{t} , context \mathbf{c} , **alternative context** $\mathbf{c}_1, \dots, \mathbf{c}_K$

$$L_{\mathbf{t},\mathbf{c}} = -\log \sigma(\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}}) - \sum_{i=1}^K \log \sigma(-\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}_i})$$

$$(a) \frac{\partial L_{\mathbf{t},\mathbf{c}}}{\partial \mathbf{u}_{\mathbf{t}}} = (\sigma(\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}}) - 1)\mathbf{v}_{\mathbf{c}} + \sum_{i=1}^K \sigma(\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}_i})\mathbf{v}_{\mathbf{c}_i}$$

$$(b) \frac{\partial L_{\mathbf{t},\mathbf{c}}}{\partial \mathbf{v}_{\mathbf{c}}} = (\sigma(\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}}) - 1)\mathbf{u}_{\mathbf{t}}$$

$$(c) \frac{\partial L_{\mathbf{t},\mathbf{c}}}{\partial \mathbf{v}_{\mathbf{c}_j}} = \sigma(\mathbf{u}_{\mathbf{t}} \cdot \mathbf{v}_{\mathbf{c}_j})\mathbf{u}_{\mathbf{t}}$$