Basics of calculus required for this document

1. $\log(a\,b\,c) = \log(a) + \log(b) + \log(c)$
2. $\log(a)^k = k\,\log(a)$
3. $\log\left(\frac{a}{b}\right) = \log(a) - \log(b)$
4. $\log(\exp\{a\}) = a$
5. $\dfrac{\mathrm{d}\log x}{\mathrm{d}x} = \dfrac{1}{x}$
6. $\dfrac{\mathrm{d}\exp\{x\}}{\mathrm{d}x} = \exp\{x\}$
7. $\dfrac{\mathrm{d}\exp\{ax\}}{\mathrm{d}x} = \exp\{ax\}\,\dfrac{\mathrm{d}ax}{\mathrm{d}x} = a\,\exp\{ax\}$
8. Derivative with respect to a vector

$$\begin{aligned} \mathcal{L} &= \mathbf{e}^T\mathbf{d} \\ \frac{\partial\mathcal{L}}{\partial\mathbf{e}} &= \mathbf{d} \end{aligned}$$

The breakdown of derivative

$$\begin{aligned} \text{Let}\,\mathbf{e} &= \begin{bmatrix} e_1 \\ e_2 \end{bmatrix} && \text{be a column vector} \\ \text{and}\,\mathbf{d} &= \begin{bmatrix} d_1 \\ d_2 \end{bmatrix} && \text{be a column vector} \\ \text{then,}\,\mathcal{L} &= \begin{bmatrix} e_1 & e_2 \end{bmatrix} \begin{bmatrix} d_1 \\ d_2 \end{bmatrix} \\ &= e_1\,d_1 \,+\, e_2\,d_2. \end{aligned}$$

Take the derivative of $\mathcal{L}$ with respect to e_1 and e_2 independently

$$\begin{aligned} \frac{\partial e_1 d_1 + e_2 d_2}{\partial e_1} &= d_1 \\ \frac{\partial e_1 d_1 + e_2 d_2}{\partial e_2} &= d_2 \end{aligned}$$

Since the derivative is w.r.t a column vector, the result should be put in a column vector

$$\frac{\partial\mathcal{L}}{\partial\mathbf{e}} = \begin{bmatrix} d_1 \\ d_2 \end{bmatrix} = \mathbf{d}.$$

Paragraph vector - distributed bag-of-words

The notation is as follows

- Let V denote the vocabulary size.
- Let N denote the number of documents.
- Let x_{ni} denote the number of occurrences of word w_i in document n.
- $\mathbf{x}_n$ implies a vector of word counts for document n.
- Training data $\mathcal{D} = \{\mathbf{x}_1, \ldots \mathbf{x}_n, \ldots, \mathbf{x}_N\}$
- Let $\mathbf{E} \in \mathbb{R}^{V \times d}$ word embeddings, where d is the dimension of embeddings, $d << V$.
- Let $\mathbf{b} \in \mathbb{R}^{V \times 1}$ denote the bias vector.
- Let $\mathbf{D} \in \mathbb{R}^{d \times N}$ N number of paragraph (document) embeddings.

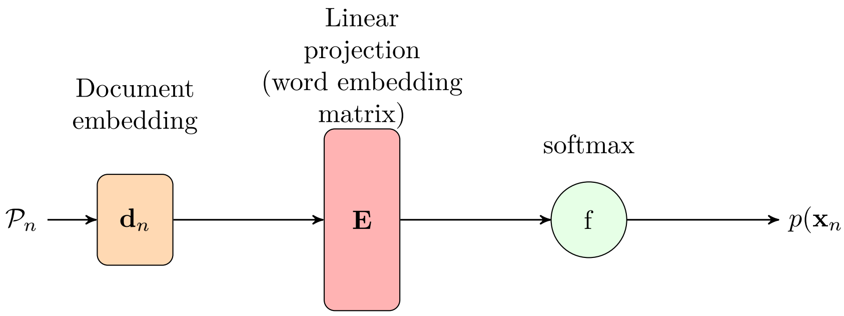

Figure 1: Paragraph vector or document model is trained to maximize the probability of all the words present in the paragraph ID $\mathcal{P}_n$.

The objective function or log-likelihood of the training data

$$\begin{aligned}
\mathcal{L} &= \sum_{n=1}^{N} \log p(\mathbf{x}_n) \\
&= \sum_{n=1}^{N} \sum_{i=1}^{V} \log(p(x_{ni}))^{x_{ni}} \\
&= \sum_{n=1}^{N} \sum_{i=1}^{V} \log(\theta_{ni})^{x_{ni}} \\
&= \sum_{n=1}^{N} \sum_{i=1}^{V} x_{ni} \log(\theta_{ni}) \\
&= \sum_{n=1}^{N} \sum_{i=1}^{V} x_{ni} \log\left[\frac{\exp\{b_i + \mathbf{e}_i^T \mathbf{d}_n\}}{\sum_{j=1}^{V} \exp\{b_i + \mathbf{e}_i^T \mathbf{d}_n\}}\right] \\
\mathcal{L} &= \sum_{n=1}^{N} \sum_{i=1}^{V} x_{ni} \Big[(b_i + \mathbf{e}_i^T \mathbf{d}_n) - \log\big(\sum_{j=1}^{V} \exp\{b_j + \mathbf{e}_j^T \mathbf{d}_n\}\big)\Big] \qquad (1)
\end{aligned}$$

Gradient w.r.t. word embedding $\nabla_{\mathbf{e}_k}\mathcal{L}$

The word embedding matrix is $\mathbf{E} \in \mathbb{R}^{V \times d}$, where each row $\mathbf{e}_k \quad k = 1 \dots V$ represents a word embedding.
Since $\mathbf{e}_k$ is a row-vector, the final gradient will also be a row-vector.

$$
\begin{aligned}
\nabla_{\mathbf{e}_k}\mathcal{L} &= \frac{\partial \sum_{n=1}^{N} \sum_{i=1}^{V} x_{ni} \Big[(b_i + \mathbf{e}_i^T \mathbf{d}_n) - \log\big(\sum_j \exp\{b_j + \mathbf{e}_j^T \mathbf{d}_n\} \big) \Big]}{\partial \mathbf{e}_k} \\
&= \sum_n \sum_i x_{ni} \Big[\frac{\partial (b_i + \mathbf{e}_i^T \mathbf{d}_n)}{\partial \mathbf{e}_k} \Big] - \sum_i x_{ni} \Big[\frac{\partial \log\big(\sum_j \exp\{b_j + \mathbf{e}_j^T \mathbf{d}_n\} \big)}{\partial \mathbf{e}_k} \Big] \\
&= \sum_n \Big[x_{nk} (\mathbf{d}_n^T) \Big] - \sum_i x_{ni} \Big[\frac{1}{\sum_j \exp\{b_j + \mathbf{e}_j^T \mathbf{d}_n\}} \big(\frac{\partial \sum_j \exp\{b_j + \mathbf{e}_j^T \mathbf{d}_n\}}{\partial \mathbf{e}_k} \big) \Big] \\
&= \sum_n \Big[x_{nk} \mathbf{d}_n^T \Big] - \sum_i x_{ni} \Big[\frac{1}{\sum_j \exp\{b_j + \mathbf{e}_j^T \mathbf{d}_n\}} \big(0 + \ldots + \exp\{b_k + \mathbf{e}_k^T \mathbf{d}_n\} \mathbf{d}_n^T \big) \Big] \\
&= \sum_n \Big[x_{nk} \mathbf{d}_n^T \Big] - \sum_i x_{ni} \Big[\underbrace{\frac{\exp\{b_i + \mathbf{e}_k^T \mathbf{d}_n^T\}}{\sum_j \exp\{b_j + \mathbf{e}_j^T \mathbf{d}_n\}}}_{\theta_{nk}} \mathbf{d}_n \Big] \\
&= \sum_n \Big[x_{nk} \mathbf{d}_n^T \Big] - \sum_i x_{in} \Big[\theta_{nk} \mathbf{d}_n^T \Big] \\
&= \sum_{n=1}^{N} \Big[x_{nk} - (\sum_{i=1}^{V} x_{ni}) \theta_{nk} \Big] \mathbf{d}_n^T
\end{aligned} \tag{2}
$$

Interpretation of the gradient (i.e., derivative of log-likelihood)

- x_{nk}: number of times word k appeared in document n.
- θ_{nk}: the estimated probability of word k in document n.
- $\sum_i x_{ni}$: sum of all the word counts in document n.
- $(\sum_i x_{ni})\theta_{nk}$: relative count of word k in document n.
- $\Big[x_{nk} - (\sum_i x_{ni})\theta_{nk}\Big]$: the difference of absolute word count to the relative word count.
- $\Big[x_{nk} - (\sum_i x_{ni})\theta_{nk}\Big]\mathbf{d}_n$, and weight this along the direction of document embedding.
- The final gradient is the sum of all weighted document embeddings.

Gradient w.r.t. document embeddings $\nabla_{\mathbf{d}_n}\mathcal{L}$

The document embedding matrix is $\mathbf{D} \in \mathbb{R}^{d \times N}$, where each column $\mathbf{d}_n \quad n = 1 \dots N$ represents a document embedding.

$$\mathcal{L} = \sum_{n=1}^{N}\sum_{i=1}^{V} x_{ni}\Big[(b_i + \mathbf{e}_i^T\mathbf{d}_n) - \log\big(\sum_{j=1}^{V}\exp\{b_j + \mathbf{e}_j^T\mathbf{d}_n\}\big)\Big]$$

$$\begin{aligned}
\nabla_{\mathbf{d}_n}\mathcal{L} &= \frac{\partial \sum_{n=1}^{N}\sum_{i=1}^{V} x_{ni}\Big[(b_i + \mathbf{e}_i^T\mathbf{d}_n) - \log\big(\sum_j \exp\{b_j + \mathbf{e}_j^T\mathbf{d}_n\}\big)\Big]}{\partial \mathbf{d}_n} \\
&= \sum_{i=1}^{V} x_{ni}\Big(0 + \mathbf{e}_i - \frac{1}{\big(\sum_j \exp\{b_j + \mathbf{e}_j^T\mathbf{d}_n\}\big)}\big(\sum_k \exp\{b_k + \mathbf{e}_k^T\mathbf{d}_n\}\,\mathbf{e}_k\big)\Big) \\
&= \sum_{i=1}^{V} x_{ni}\Big(\mathbf{e}_i - \sum_k \mathbf{e}_k\,\underbrace{\frac{\exp\{b_k + \mathbf{e}_k^T\mathbf{d}_n\}}{\sum_j \exp\{b_j + \mathbf{e}_j^T\mathbf{d}_n\}}}_{\theta_{nk}}\Big) \\
&= \sum_{i=1}^{V} x_{ni}\Big(\mathbf{e}_i - (\sum_{k=1}^{V}\mathbf{e}_k\,\theta_{nk})\Big) \\
&= \Big(\sum_{i=1}^{V}\mathbf{e}_i x_{ni}\Big) - \Big((\sum_{k=1}^{V}\mathbf{e}_k\theta_{nk})(\sum_{i=1}^{V} x_{ni})\Big) \\
&= \sum_{i=1}^{V}\mathbf{e}_i\Big(x_{ni} - \theta_{ni}(\sum_{k=1}^{V} x_{nk})\Big) \qquad (3)
\end{aligned}$$

Interpretation of the gradient (i.e., derivative of log-likelihood)

- x_{ni}: number of times word i appeared in document n.
- θ_{ni}: the estimated probability of word i in document n.
- $\sum_k x_{nk}$: sum of all the word counts in document n.
- $\theta_{ni}(\sum_i x_{ni})$: relative count of word i in document n.
- $\Big[x_{nk} - (\sum_i x_{ni})\theta_{nk}\Big]$: the difference of absolute word count to the relative word count.
- $\mathbf{e}_i\Big[x_{nk} - (\sum_i x_{ni})\theta_{nk}\Big]$: weight this along the direction of word embedding for word i.
- The final gradient is the sum of all weighted word embeddings.

Training the model

Algorithm 1 Training algorithm

Require: Training data $\mathbf{x}_1 \dots \mathbf{x}_n$
Require: Vocabulary of size V
Initialize $\mathbf{E}, \mathbf{b}, \mathbf{D}$ to small random values sampling from $\mathcal{N}(0, 0.001)$
$\eta = 0.1$ ▷ learning rate
Initialize bias vector $b_i = \log\left(\frac{\sum_{n=1}^{N} x_{ni}}{\sum_{n=1}^{N}\sum_{i=1}^{V} x_{ni}}\right)$
for $i = 0; i < 100; i++$ **do** ▷ Training iterations
  for $k = 0; k < V; k++$ **do**
    Compute gradient $\nabla_{\mathbf{e}_k}\mathcal{L}$ using Eq. (2)
    $\mathbf{e}_k \leftarrow \mathbf{e}_k + \eta\nabla_{\mathbf{e}_k}\mathcal{L}$ ▷ Update word embedding $\mathbf{e}_k$
  end for
  for $n = 1; n < N; n++$ **do**
    Compute gradient $\nabla_{\mathbf{d}_n}\mathcal{L}$ using Eq. (3)
    $\mathbf{d}_n \leftarrow \mathbf{d}_n + \eta\nabla_{\mathbf{d}_n}\mathcal{L}$ ▷ Update document embedding $\mathbf{d}_n$
  end for
end for

Inference

To obtain document embeddings for a test sentence or document.

Algorithm 2 Inference algorithm

Require: Test document $\mathbf{x}_t$ ▷ Vector of word counts
Use trained $\mathbf{E}, \mathbf{b}$
Initialize $\mathbf{d}_t$ to small random values sampling from $\mathcal{N}(0, 0.001)$
$\eta = 0.1$ ▷ learning rate
for $i = 0; i < 20; i++$ **do** ▷ Inference iterations
  Compute gradient $\nabla_{\mathbf{d}_t}\mathcal{L}$ using Eq. (3)
  $\mathbf{d}_t \leftarrow \mathbf{d}_t + \eta\nabla_{\mathbf{d}_t}\mathcal{L}$ ▷ Update document embedding $\mathbf{d}_t$
end for

Gradient computation by hand

Consider the following toy example where we have $N = 2$ documents and vocabulary of size $V = 3$.

Document	**Word index** $\rightarrow$		
index $\downarrow$	w_1	w_2	w_3
1	5	3	0
2	1	2	3

The document embeddings (each column in one doc embedding) are

$$\mathbf{D} = \begin{bmatrix} \mathbf{d}_1 & \mathbf{d}_2 \\ 2.5 & 1.0 \\ -2.0 & 3.0 \end{bmatrix}$$

The estimated probabilities are

$$\boldsymbol{\theta} = \begin{bmatrix} 0.625 & 0.375 & 0.0 \\ 0.1 & 0.3 & 0.6 \end{bmatrix}$$

Compute gradients of the objective w.r.t word embeddings using the following equation

$$\nabla_{\mathbf{e}} \mathcal{L} = \sum_{n=1}^{N} \Big[x_{nk} - (\sum_{i=1}^{V} x_{ni}) \theta_{nk} \Big] \mathbf{d}_n^T$$

1. $\nabla_{\mathbf{e}_1} \mathcal{L} =$

2. $\nabla_{\mathbf{e}_2} \mathcal{L} =$

3. $\nabla_{\mathbf{e}_3} \mathcal{L} =$

Hands on exercise jupyter notebook

Here is the link to Google collab notebook
`https://colab.research.google.com/drive/1RFeAoiYICGh4R31x_gO38IiinTGgixDD?usp=sharing`