



WORD EMBEDDING

**Mathematics Foundations for Computer Science
PhD. Nguyễn An Khương**

Students:

1. Nguyễn Hữu Trưởng - 2470573
2. Cao Nguyễn Minh Hiếu - 2470575
3. Võ Thị Bích Phượng - 2470570
4. Huch Sreyneang - 2470902
5. Hoàng Thế Sơn - 2470742

Ho Chi Minh City, May - 2025



Contents

- 1. Introduction**
- 2. Skip-gram**
- 3. Continuous Bag of Words (CBOW)**
- 4. Exercises**
- 5. Demo**

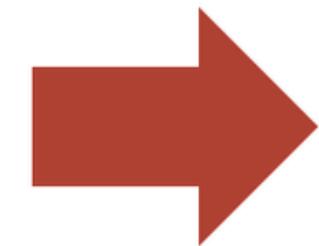
1. Introduction



1. Introduction

- Trong xử lý ngôn ngữ tự nhiên (NLP), bước đầu tiên và quan trọng là biểu diễn từ ngữ dưới dạng số để đưa vào mô hình học máy.
- Một phương pháp trực quan và phổ biến là vector one-hot, trong đó mỗi từ được biểu diễn bằng một vector nhị phân có độ dài bằng kích thước từ điển, với đúng một phần tử bằng 1 và các phần tử còn lại bằng 0.

Vocabulary:
Man, woman, boy,
girl, prince,
princess, queen,
king, monarch



	1	2	3	4	5	6	7	8	9
man	1	0	0	0	0	0	0	0	0
woman	0	1	0	0	0	0	0	0	0
boy	0	0	1	0	0	0	0	0	0
girl	0	0	0	1	0	0	0	0	0
prince	0	0	0	0	1	0	0	0	0
princess	0	0	0	0	0	1	0	0	0
queen	0	0	0	0	0	0	1	0	0
king	0	0	0	0	0	0	0	1	0
monarch	0	0	0	0	0	0	0	0	1

Each word gets
a 1x9 vector
representation

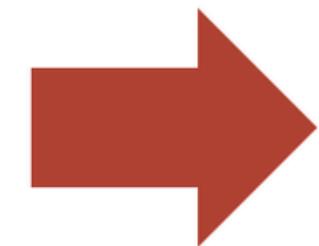


1. Introduction

Mặc dù đơn giản và dễ triển khai, biểu diễn one-hot tồn tại những hạn chế nghiêm trọng:

- Không có thông tin ngữ nghĩa
- Độ thưa (Sparsity) cao
- Không học được quan hệ ngữ cảnh

Vocabulary:
Man, woman, boy,
girl, prince,
princess, queen,
king, monarch



	1	2	3	4	5	6	7	8	9
man	1	0	0	0	0	0	0	0	0
woman	0	1	0	0	0	0	0	0	0
boy	0	0	1	0	0	0	0	0	0
girl	0	0	0	1	0	0	0	0	0
prince	0	0	0	0	1	0	0	0	0
princess	0	0	0	0	0	1	0	0	0
queen	0	0	0	0	0	0	1	0	0
king	0	0	0	0	0	0	0	1	0
monarch	0	0	0	0	0	0	0	0	1

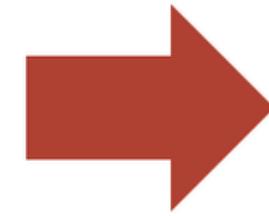
Each word gets
a 1x9 vector
representation



1. Introduction

- Word embedding là kỹ thuật học biểu diễn các từ dưới dạng vector liên tiếp có **số chiều thấp**, sao cho các vector này **mang thông tin ngữ nghĩa và ngữ cảnh**.
- Khác với vector one-hot, các vector embedding được học từ dữ liệu, nhờ đó:
 - Các từ có ý nghĩa gần nhau sẽ có vector gần nhau.
 - Có thể thực hiện các phép so sánh như: king - queen = man - woman

Vocabulary:
Man, woman, boy,
girl, prince,
princess, queen,
king, monarch



	Femininity	Youth	Royalty
Man	0	0	0
Woman	1	0	0
Boy	0	1	0
Girl	1	1	0
Prince	0	1	1
Princess	1	1	1
Queen	1	0	1
King	0	0	1
Monarch	0.5	0.5	1

Each word gets a
1x3 vector

Similar words...
similar vectors



1. Introduction

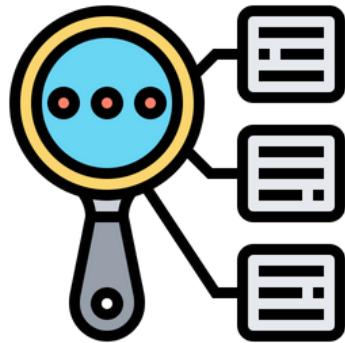
- **Ứng dụng:** Word embedding là nền tảng cốt lõi của hầu hết các mô hình NLP hiện đại



Text retrieval



Machine Translation



Text Classification



Chatbot

- Phạm vi trình bày: Hai mô hình tiêu biểu khởi đầu của Word embedding:

- **Skip-gram**
- **Continuous Bag of Words (CBOW)**

2. Skip-gram

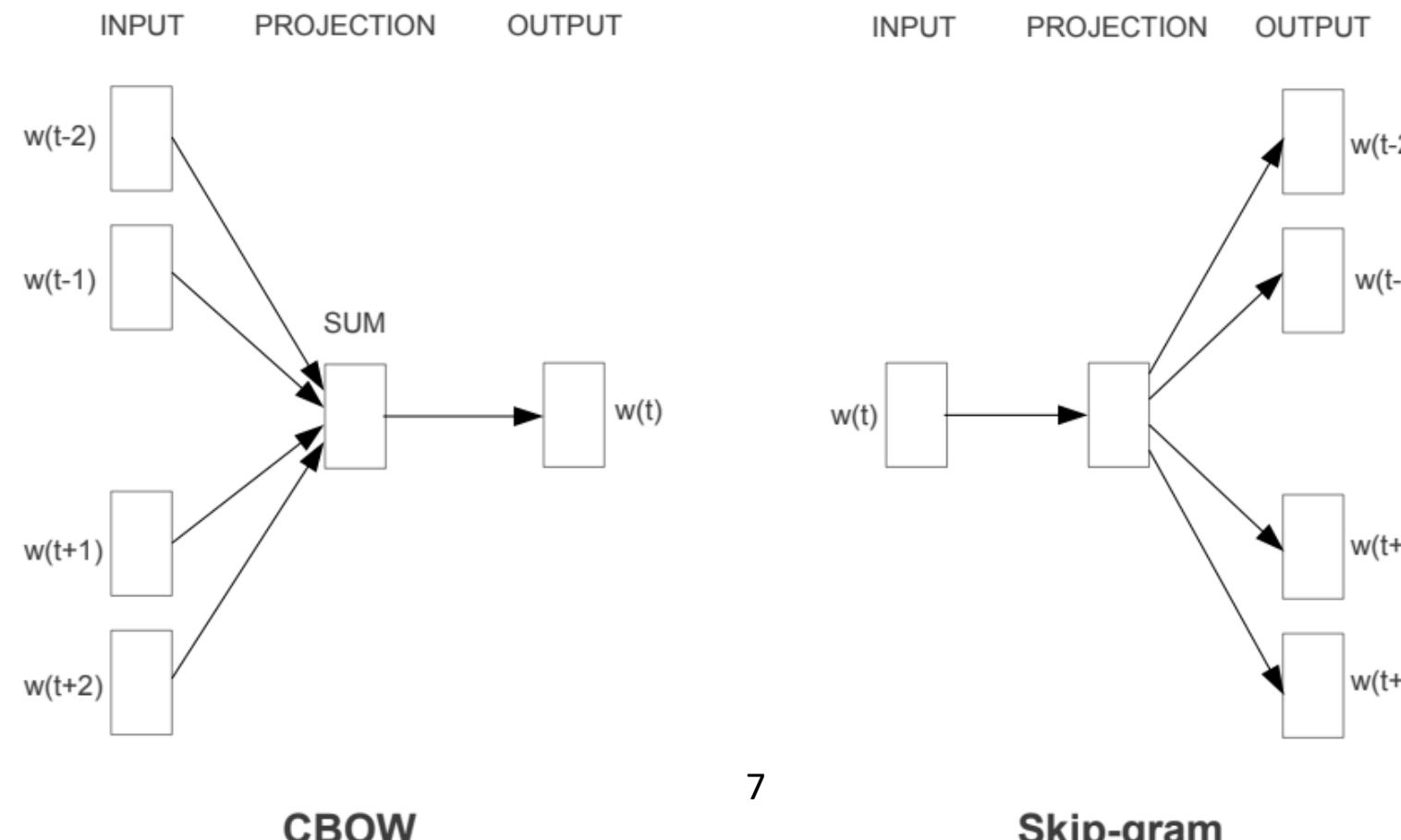
2. Skip-gram

2.1. Giới thiệu

Word2vec là gì? Word2vec là một kỹ thuật để học word embeddings từ một lượng lớn dữ liệu văn bản mà không cần dữ liệu đã được gắn nhãn (như phân loại từ); thay vào đó, nó sử dụng self-supervised learning, nghĩa là nó tự học từ chính dữ liệu bằng cách dự đoán một phần dữ liệu dựa trên phần khác.

Word2vec có hai mô hình chính:

- **Skip-gram:** Skip-gram dự đoán các từ ngữ cảnh dựa trên từ mục tiêu. Với từ "sits", mô hình cố gắng dự đoán các từ như "the", "cat", "on", "the", "mat".
- **Continuous Bag of Words (CBOW):** Ngược lại với Skip-gram, dự đoán một từ mục tiêu dựa trên các từ ngữ cảnh xung quanh nó. Ví dụ, với câu "the cat sits on the mat", CBOW lấy các từ "the", "cat", "on", "the", "mat" để dự đoán từ "sits".

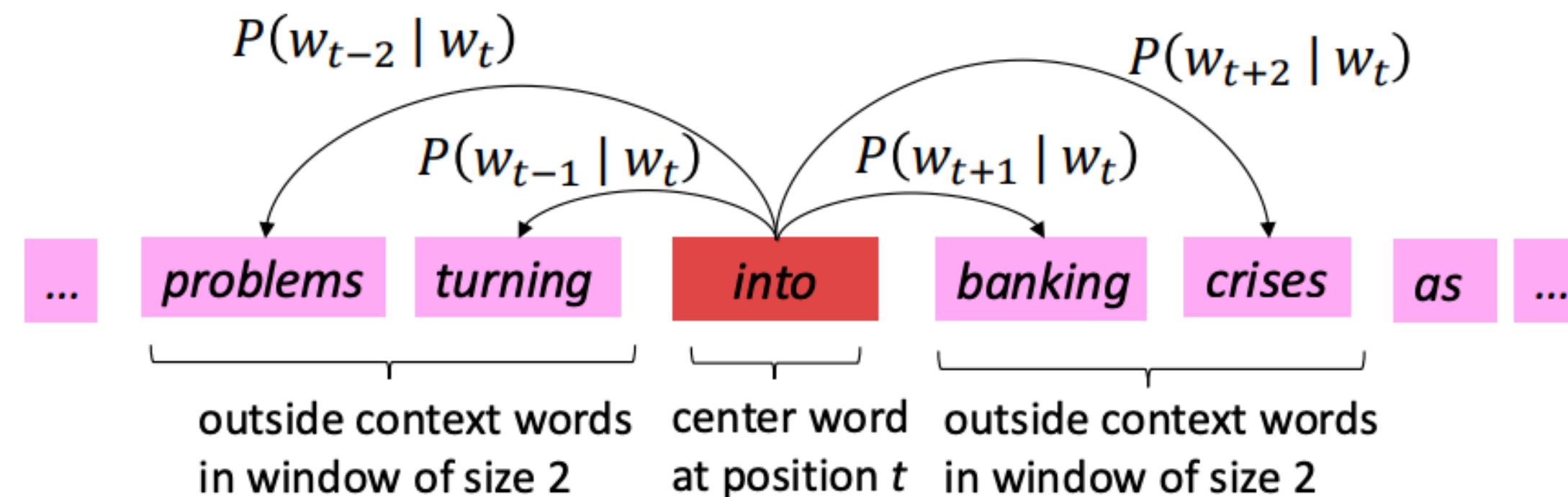


2. Skip-gram

2.2. Skip-gram

Mô hình skip-gram giả định rằng một từ có thể được sử dụng để sinh ra các từ xung quanh nó trong một chuỗi văn bản.

Ta giả định rằng, với từ đích trung tâm cho trước, các từ ngữ cảnh được sinh ra độc lập với nhau, mô hình skip-gram quan tâm đến xác suất có điều kiện sinh ra các từ ngữ cảnh nằm trong khoảng cách không quá window size.





2. Skip-gram

2.2. Skip-gram

Trong mô hình skip-gam, mỗi từ được biểu diễn bằng hai vector d – chiều (một dùng khi từ w là từ ngữ cảnh, một dùng khi từ w là từ trung tâm) để tính xác suất có điều kiện. Sử dụng hàm softmax trên tích vô hướng của vector để tính xác suất có điều kiện sinh ra từ ngữ cảnh cho một từ đích trung tâm cho trước:

$$P(w_o | w_c) = \frac{\exp(\mathbf{u}_o^\top \mathbf{v}_c)}{\sum_{i \in \mathcal{V}} \exp(\mathbf{u}_i^\top \mathbf{v}_c)},$$

Trong đó:

- w_c : Từ trung tâm (center word), ví dụ "cat" trong câu "the cat sat on mat".
- w_o : Từ ngữ cảnh (output word), ví dụ "sat" trong cùng câu.
- $\mathbf{v}_c \in \mathbb{R}^d$: Vector biểu diễn của từ trung tâm w_c (input vector).
- $\mathbf{u}_o \in \mathbb{R}^d$: Vector biểu diễn của từ ngữ cảnh w_o (output vector).
- \mathcal{V} : Tập hợp tất cả từ trong từ điển (vocabulary). Tập chỉ số trong bộ từ vựng là $\mathcal{V} = \{0, 1, \dots, |\mathcal{V}| - 1\}$.
- $\mathbf{u}_i \in \mathbb{R}^d$: Vector biểu diễn của từ i trong từ điển.
- \exp : Hàm mũ, $\exp(x) = e^x$
- $\mathbf{u}_o^\top \mathbf{v}_c$: Tích vô hướng (dot product) giữa vector \mathbf{u}_o và \mathbf{v}_c , đo lường mức độ tương đồng giữa từ trung tâm và từ ngữ cảnh.
- $\sum_{i \in \mathcal{V}} \exp(\mathbf{u}_i^\top \mathbf{v}_c)$: Tổng chuẩn hóa (normalization term) giữa các điểm tương đồng mũ giữa từ trung tâm w_c trên toàn bộ từ điển \mathcal{V} , đảm bảo các xác suất cộng lại bằng 1.



2. Skip-gram

2.3. Hàm hợp lý (likelihood)

Giả sử ta có một chuỗi văn bản với độ dài T , trong đó từ tại vị trí t được ký hiệu là $w^{(t)}$. Mô hình skip-gram giả định rằng:

- Các từ ngữ cảnh trong cửa sổ kích thước m (bao gồm m từ bên trái và m từ bên phải của từ trung tâm) được sinh ra **độc lập** với nhau, khi biết từ đích trung tâm.
- Hàm hợp lý (likelihood) của mô hình là xác suất kết hợp để sinh ra tất cả các từ ngữ cảnh cho mọi từ trung tâm trong chuỗi văn bản.

Hàm hợp lý được định nghĩa như sau:

$$L = \prod_{t=1}^T \prod_{-m \leq j \leq m, j \neq 0} P(w^{(t+j)} | w^{(t)})$$

Nếu $t + j < 1$ hoặc $t + j > T$ (vị trí ngoài chuỗi văn bản), các số hạng này bị bỏ qua. Ví dụ, với $t = 1$ và $m = 2$, các vị trí $j = -2, -1$ sẽ không tồn tại và được bỏ qua.

Ý nghĩa của hàm hợp lý:

- Hàm L biểu thị xác suất tổng quát để mô hình skip-gram sinh ra toàn bộ các cặp từ trung tâm và từ ngữ cảnh trong tập dữ liệu.
- Mục tiêu của MLE là tìm các tham số $(\mathbf{v}_i, \mathbf{u}_i)$ sao cho L đạt giá trị lớn nhất, tức là mô hình dự đoán các từ ngữ cảnh chính xác nhất có thể.



2. Skip-gram

2.4. Trainning

Mục tiêu của skip-gram là cực đại hóa hàm hợp lý (Maximum Likelihood Estimation - MLE), tức là tìm các vector \mathbf{v}_i và \mathbf{u}_i để cho L đạt giá trị lớn nhất. Hay mô hình có thể dự đoán chính xác các từ ngữ cảnh dựa trên từ trung tâm.

Trong thực tế, vì L là một tích của nhiều xác suất nhỏ, ta thường làm việc với log-likelihood để tránh vấn đề số học (numerical underflow):

$$\log L = \sum_{t=1}^T \sum_{-m \leq j \leq m, j \neq 0} \log P(w^{(t+j)} | w^{(t)}).$$

Hàm mất mát được định nghĩa là phủ định của log-likelihood:

$$J = -\log L = -\sum_{t=1}^T \sum_{-m \leq j \leq m, j \neq 0} \log P(w^{(t+j)} | w^{(t)}).$$



$$\begin{aligned}\log P(w_o | w_c) &= \log \frac{\exp(\mathbf{u}_o^\top \mathbf{v}_c)}{\sum_{i \in \mathcal{V}} \exp(\mathbf{u}_i^\top \mathbf{v}_c)} \\ &= \log \exp(\mathbf{u}_o^\top \mathbf{v}_c) - \log \left(\sum_{i \in \mathcal{V}} \exp(\mathbf{u}_i^\top \mathbf{v}_c) \right) \\ &= \mathbf{u}_o^\top \mathbf{v}_c - \log \left(\sum_{i \in \mathcal{V}} \exp(\mathbf{u}_i^\top \mathbf{v}_c) \right)\end{aligned}$$



2. Skip-gram

2.4. Trainning

Ta cần tính đạo hàm của $\log P(w_o \mid w_c)$ theo \mathbf{v}_c :

$$\begin{aligned}\frac{\partial}{\partial \mathbf{v}_c} \log P(w_o \mid w_c) &= \frac{\partial}{\partial \mathbf{v}_c} \left(\mathbf{u}_o^\top \mathbf{v}_c - \log \left(\sum_{i \in \mathcal{V}} \exp(\mathbf{u}_i^\top \mathbf{v}_c) \right) \right) \\ &= \frac{\partial}{\partial \mathbf{v}_c} \mathbf{u}_o^\top \mathbf{v}_c - \frac{\partial}{\partial \mathbf{v}_c} \log \left(\sum_{i \in \mathcal{V}} \exp(\mathbf{u}_i^\top \mathbf{v}_c) \right) \\ &= \mathbf{u}_o - \frac{1}{\sum_{i \in \mathcal{V}} \exp(\mathbf{u}_i^\top \mathbf{v}_c)} \cdot \sum_{j \in \mathcal{V}} \exp(\mathbf{u}_j^\top \mathbf{v}_c) \mathbf{u}_j\end{aligned}$$

Nhận thấy:

$$\frac{\exp(\mathbf{u}_j^\top \mathbf{v}_c)}{\sum_{i \in \mathcal{V}} \exp(\mathbf{u}_i^\top \mathbf{v}_c)} = P(w_j \mid w_c),$$

nên:

$$\frac{1}{\sum_{i \in \mathcal{V}} \exp(\mathbf{u}_i^\top \mathbf{v}_c)} \cdot \sum_{j \in \mathcal{V}} \exp(\mathbf{u}_j^\top \mathbf{v}_c) \mathbf{u}_j = \sum_{j \in \mathcal{V}} P(w_j \mid w_c) \mathbf{u}_j.$$

Do đó, gradient tổng quát là:

$$\frac{\partial \log P(w_o \mid w_c)}{\partial \mathbf{v}_c} = \mathbf{u}_o - \sum_{j \in \mathcal{V}} P(w_j \mid w_c) \mathbf{u}_j.$$

Gradient này được sử dụng để cập nhật \mathbf{v}_c trong SGD theo công thức:

$$\mathbf{v}_c \leftarrow \mathbf{v}_c + \eta \left(\mathbf{u}_o - \sum_{j \in \mathcal{V}} P(w_j \mid w_c) \mathbf{u}_j \right),$$

với η là tốc độ học (learning rate).

3. Continuous Bag of Words (CBOW)



3. Continuous Bag of Words (CBOW)

3.1. Giới thiệu về mô hình



Mục tiêu: Dự đoán từ trung tâm (target word) dựa trên các từ xung quanh (context words).

Các ký hiệu:

Giả sử chúng ta có một tập từ vựng V với kích thước $|V|$.

Mỗi từ được biểu diễn bằng một **vector one-hot** kích thước $|V|$:

- Nếu từ w_i là từ thứ i trong tập từ vựng, thì vector one-hot là:

$$\mathbf{x}_i = [0 \quad 0 \quad \dots \quad 1 \quad \dots \quad 0]^T \quad (\text{vị trí thứ } i \text{ là } 1)$$

Giả sử:

- Cửa sổ ngữ cảnh có kích thước $2c$, tức là ta nhìn c từ trái và c từ phải của từ trung tâm w_t .
- Mỗi từ ngữ cảnh được ký hiệu là w_{t-j} và w_{t+j} với $j = 1, 2, \dots, c$.



3. Continuous Bag of Words (CBOW)

3.2. Cấu trúc mạng của CBOW

Lớp đầu vào: Gồm 2cvector one-hot : $\mathbf{x}_{t-c}, \dots, \mathbf{x}_{t-1}, \mathbf{x}_{t+1}, \dots, \mathbf{x}_{t+c}$

Lớp ẩn: Với vector one-hot \mathbf{x} , thì embedding vector là: $\mathbf{v} = \mathbf{W}^T \mathbf{x}$

Ta lấy trung bình của các vector embedding từ ngữ cảnh: $\mathbf{h} = \frac{1}{2c} \sum_{-c \leq j \leq c, j \neq 0} \mathbf{W}^T \mathbf{x}_{t+j}$

Lớp đầu ra: Có ma trận trọng số $\mathbf{W}' \in \mathbb{R}^{N \times |V|}$

, ánh xạ từ không gian nhúng về xác suất phân phối trên từ vựng.

Tích tích vô hướng: $\mathbf{u} = \mathbf{W}'^T \mathbf{h} \in \mathbb{R}^{|V|}$

Tính xác suất từ trung tâm là từ w_t : $p(w_t | \text{context}) = \frac{\exp(u_t)}{\sum_{i=1}^{|V|} \exp(u_i)}$



3. Continuous Bag of Words (CBOW)

3.3. Hàm mất mát (Loss function)

Mục tiêu là tối đa hóa xác suất dự đoán đúng từ trung tâm w_t

Hàm mất mát là hàm cross-entropy (âm log-likelihood):

$$\mathcal{L} = -\log p(w_t | \text{context}) = -\log \left(\frac{\exp(u_t)}{\sum_{i=1}^{|V|} \exp(u_i)} \right) = -u_t + \log \left(\sum_{i=1}^{|V|} \exp(u_i) \right)$$

Trọng số \mathbf{W} và \mathbf{W}' được học bằng gradient descent hoặc các biến thể như SGD.

Vì softmax có mẫu số lớn (sum qua toàn bộ từ vựng), có thể dùng Negative Sampling để tăng hiệu suất tính toán.



3. Continuous Bag of Words (CBOW)

3.4. Ví dụ minh họa

Câu 1: "The quick brown fox jumps over the lazy dog"

Từ trung tâm: "jumps"

Ngữ cảnh (với $c=2$): "brown", "fox", "over", "the"

-> CBOW học biểu diễn sao cho từ "jumps" có xác suất cao nhất khi biết 4 từ ngữ cảnh trên.

Câu 2: "The cat sits on the mat"

Với cửa sổ ngữ cảnh $c=2$, nếu từ trung tâm là "sits", thì ngữ cảnh là:

- Trái: "The", "cat"
- Phải: "on", "the"



3. Continuous Bag of Words (CBOW)

3.5. Sơ đồ dòng dữ liệu

Tầng đầu vào (one-hot encoding)

4 từ ngữ cảnh được chuyển sang vector one-hot kích thước $|V|$

x_1	x_2	x_3	x_4
[0...1...0]	[0...1...0]	[0...1...0]	[0...1...0]

(Kích thước $|V|$)

Nhúng từ (embedding)

Mỗi one-hot vector nhân với ma trận $\mathbf{W} \in \mathbb{R}^{|V| \times N}$ để lấy embedding $v_i = \mathbf{W}^T x_i \in \mathbb{R}^N$

v_1	v_2	v_3	v_4
[0.2 ... 0.1]

Trung bình các vector $h = \frac{1}{4}(v_1 + v_2 + v_3 + v_4)$ $h = [0.3, 0.5, \dots, 0.1] \in \mathbb{R}^N$

Tầng đầu ra (dự đoán từ trung tâm)

Nhân với ma trận $\mathbf{W}' \in \mathbb{R}^{N \times |V|}$: $u = \mathbf{W}'^T h \in \mathbb{R}^{|V|}$

$u = [1.2, 0.5, 3.1, \dots, 0.8]$

$p(w_t | context)$ = xác suất trên toàn từ vựng



3. Continuous Bag of Words (CBOW)

3.6. Training

Từ hàm hợp lý của mô hình CBOW:

$$L = \prod_{t=1}^T P(w^{(t)} | w^{(t-m)}, \dots, w^{(t-1)}, w^{(t+1)}, \dots, w^{(t+m)}).$$

Ta cần tìm ước lượng hợp lý cực đại của mô hình CBOW bằng cách cực tiểu hóa hàm mất mát (tương tự như Skip-gram).

Để việc tối ưu diễn ra thuận tiện hơn, ta lấy logarith của hàm hợp lý để biến tích thành tổng:

$$\log(L) = \sum_{t=1}^T \log P(w^{(t)} | w^{(t-m)}, \dots, w^{(t-1)}, w^{(t+1)}, \dots, w^{(t+m)}).$$

Sau đó lấy phủ định để có được hàm mất mát:

$$J = -\log(L) = -\sum_{t=1}^T \log P(w^{(t)} | w^{(t-m)}, \dots, w^{(t-1)}, w^{(t+1)}, \dots, w^{(t+m)}).$$

Để thuận tiện cho việc tính toán, ta sử dụng logarith của hàm **xác suất sinh ra từ đích trung tâm dựa vào các từ ngữ cảnh cho trước**:

$$\begin{aligned} \log P(w_c | \mathcal{W}_o) &= \log \frac{\exp(\mathbf{u}_c^\top \bar{\mathbf{v}}_o)}{\sum_{i \in \mathcal{V}} \exp(\mathbf{u}_i^\top \bar{\mathbf{v}}_o)} \\ &= \log (\exp(\mathbf{u}_c^\top \bar{\mathbf{v}}_o)) - \log \left(\sum_{i \in \mathcal{V}} \exp(\mathbf{u}_i^\top \bar{\mathbf{v}}_o) \right) \\ &= \mathbf{u}_c^\top \bar{\mathbf{v}}_o - \log \left(\sum_{i \in \mathcal{V}} \exp(\mathbf{u}_i^\top \bar{\mathbf{v}}_o) \right). \end{aligned}$$



3. Continuous Bag of Words (CBOW)

3.6. Training

Để tìm gradient của hàm mất mát, ta lấy đạo hàm của phương trình trên theo từng vector ngữ cảnh \mathbf{v}_{o_i} ($i = 1, \dots, 2m$):

$$\frac{\partial \log P(w_c | \mathcal{W}_o)}{\partial \mathbf{v}_{o_i}} = \frac{\partial}{\partial \mathbf{v}_{o_i}} (\mathbf{u}_c^\top \bar{\mathbf{v}}_o) - \frac{\partial}{\partial \mathbf{v}_{o_i}} \left(\log \left(\sum_{j \in \mathcal{V}} \exp(\mathbf{u}_j^\top \bar{\mathbf{v}}_o) \right) \right)$$

Hãy tính hạng tử đầu tiên:

$$\frac{\partial}{\partial \mathbf{v}_{o_i}} (\mathbf{u}_c^\top \bar{\mathbf{v}}_o) = \frac{1}{2m} \mathbf{u}_c^\top \frac{\partial}{\partial \mathbf{v}_{o_i}} \left(\sum_{i=1}^{2m} \mathbf{v}_{o_i} \right) = \frac{1}{2m} \mathbf{u}_c$$

Và hạng tử thứ hai:

$$\frac{\partial}{\partial \mathbf{v}_{o_i}} \left(\log \left(\sum_{j \in \mathcal{V}} \exp(\mathbf{u}_j^\top \bar{\mathbf{v}}_o) \right) \right) = \frac{1}{\sum_{i \in \mathcal{V}} \exp(\mathbf{u}_i^\top \bar{\mathbf{v}}_o)} \left(\frac{\partial}{\partial \mathbf{v}_{o_i}} \left(\sum_{j \in \mathcal{V}} \exp(\mathbf{u}_j^\top \bar{\mathbf{v}}_o) \right) \right)$$

Với:

$$\frac{\partial}{\partial \mathbf{v}_{o_i}} \exp(\mathbf{u}_j^\top \bar{\mathbf{v}}_o) = \exp(\mathbf{u}_j^\top \bar{\mathbf{v}}_o) \frac{\partial}{\partial \mathbf{v}_{o_i}} (\mathbf{u}_j^\top \bar{\mathbf{v}}_o) = \exp(\mathbf{u}_j^\top \bar{\mathbf{v}}_o) \frac{1}{2m} \mathbf{u}_j$$

Ghép vào hạng tử thứ hai ta có:

$$\frac{1}{\sum_{i \in \mathcal{V}} \exp(\mathbf{u}_i^\top \bar{\mathbf{v}}_o)} \left(\sum_{j \in \mathcal{V}} \exp(\mathbf{u}_j^\top \bar{\mathbf{v}}_o) \frac{1}{2m} \mathbf{u}_j \right) = \frac{1}{2m} \sum_{j \in \mathcal{V}} \frac{\exp(\mathbf{u}_j^\top \bar{\mathbf{v}}_o)}{\sum_{i \in \mathcal{V}} \exp(\mathbf{u}_i^\top \bar{\mathbf{v}}_o)} \mathbf{u}_j$$

Kết hợp các hạng tử, ta có phương trình gradient tổng quát:

$$\begin{aligned} \frac{\partial \log P(w_c | \mathcal{W}_o)}{\partial \mathbf{v}_{o_i}} &= \frac{1}{2m} \mathbf{u}_c - \frac{1}{2m} \sum_{j \in \mathcal{V}} \frac{\exp(\mathbf{u}_j^\top \bar{\mathbf{v}}_o)}{\sum_{i \in \mathcal{V}} \exp(\mathbf{u}_i^\top \bar{\mathbf{v}}_o)} \mathbf{u}_j \\ &= \frac{1}{2m} \left(\mathbf{u}_c - \sum_{j \in \mathcal{V}} P(w_j | \mathcal{W}_o) \mathbf{u}_j \right) \end{aligned}$$

Gradient này được sử dụng để cập nhật \mathbf{v}_{o_i} trong SGD theo công thức:

$$\mathbf{v}_{o_i} \leftarrow \mathbf{v}_{o_i} + \eta \cdot \frac{1}{2m} \left(\mathbf{u}_c - \sum_{j \in \mathcal{V}} P(w_j | \mathcal{W}_o) \mathbf{u}_j \right),$$

với η là tốc độ học (learning rate).

4. Exercises



4. Exercises

Câu 1: Độ phức tạp tính toán của mỗi gradient là bao nhiêu? Nếu từ điển chứa một lượng lớn các từ, điều này sẽ gây ra vấn đề gì?

Công thức tính gradient Skip-gram (15.1.8):

$$\begin{aligned}\frac{\partial \log P(w_o | w_c)}{\partial \mathbf{v}_c} &= \mathbf{u}_o - \frac{\sum_{j \in \square} \exp(\mathbf{u}_j^\top \mathbf{v}_c) \mathbf{u}_j}{\sum_{i \in \square} \exp(\mathbf{u}_i^\top \mathbf{v}_c)} \\ &= \mathbf{u}_o - \sum_{j \in \square} \left(\frac{\exp(\mathbf{u}_j^\top \mathbf{v}_c)}{\sum_{i \in \square} \exp(\mathbf{u}_i^\top \mathbf{v}_c)} \right) \mathbf{u}_j \\ &= \mathbf{u}_o - \sum_{j \in \square} P(w_j | w_c) \mathbf{u}_j.\end{aligned}$$

Công thức tính gradient CBOW (15.1.15):

$$\frac{\partial \log P(w_c | \square_o)}{\partial \mathbf{v}_{o_i}} = \frac{1}{2m} \left(\mathbf{u}_c - \sum_{j \in \square} \frac{\exp(\mathbf{u}_j^\top \bar{\mathbf{v}}_o) \mathbf{u}_j}{\sum_{i \in \square} \exp(\mathbf{u}_i^\top \bar{\mathbf{v}}_o)} \right) = \frac{1}{2m} \left(\mathbf{u}_c - \sum_{j \in \square} P(w_j | \square_o) \mathbf{u}_j \right)$$

- Ở cả 2 mô hình, tính được gradient đòi hỏi tính xác suất có điều kiện cho mọi từ có trong từ điển V
- Do đó, độ phức tạp tính toán của mỗi gradient trong cả 2 TH đều là $O(|V| * d)$, với d là số chiều của vector embedding và V là từ điển
- **Vấn đề:** chậm hội tụ \Rightarrow **Giải pháp:** Negative Sampling (15.2.1)



4. Exercises

Câu 2: Có một số cụm từ cố định trong tiếng Anh bao gồm nhiều từ, chẳng hạn như “new york”. Bạn sẽ huấn luyện các vector từ của chúng như thế nào? Gợi ý: Xem phần 4 trong bài báo Word2vec[2].

- Ý tưởng: Nhận diện và tạo cụm từ (phrases) trước khi huấn luyện
 - Ví dụ: “new york” → “new_york”, “san francisco” → “san_francisco”

$$\text{score}(w_i, w_j) = \frac{C(w_i w_j) - \delta}{C(w_i) \cdot C(w_j)}$$

Trong đó:

- $C(w_i), C(w_j)$: tần suất của từng từ riêng lẻ,
- $C(w_i w_j)$: tần suất của cụm 2 từ xuất hiện liên tiếp,
- δ : một hằng số để tránh nối những cụm có tần suất thấp (thường là 5).

Đồng thời đặt ra một ngưỡng cố định. Nếu score vượt ngưỡng này, ta xem $w_i w_j$ là cụm cố định.



4. Exercises

Câu 2: Có một số cụm từ cố định trong tiếng Anh bao gồm nhiều từ, chẳng hạn như “new york”. Bạn sẽ huấn luyện các vector từ của chúng như thế nào? Gợi ý: Xem phần 4 trong bài báo Word2vec[2].

$$\text{score}(w_i, w_j) = \frac{C(w_i w_j) - \delta}{C(w_i) \cdot C(w_j)}$$

Ví dụ 1:

"new" xuất hiện 5000 lần, "york" 3000 lần, "new york" xuất hiện 2800 lần:

$$\text{score}("new", "york") = \frac{2800-5}{5000 \cdot 3000} \approx 0.000186$$

Ngưỡng là 0.0001 → "new york" được giữ lại thành "new_york".

Ví dụ 2:

"this" xuất hiện 10000 lần, "is" 9500 lần, "this is" xuất hiện 5000 lần:

$$\text{score}("this", "is") = \frac{5000-5}{10000 \cdot 9500} \approx 0.0000525789474$$

Ngưỡng là 0.0001 → "this is" không được nối lại thành cụm từ



4. Exercises

Câu 3: Sử dụng mô hình Skip-gram. Mỗi quan hệ giữa tích vô hướng của hai vector từ và độ tương tự cosin trong mô hình skip-gam là gì? Đối với một cặp từ có ngữ nghĩa gần nhau, tại sao hai vector từ này lại thường có độ tương tự cosin cao?

- Nhắc lại công thức tích vô hướng:

$$\mathbf{u}_{w_o}^\top \mathbf{v}_{w_c} = \|\mathbf{u}_{w_o}\| \cdot \|\mathbf{v}_{w_c}\| \cdot \cos(\theta)$$

- Nhắc lại công thức Skip-gram

$$P(w_o|w_c) = \frac{\exp(\mathbf{u}_{w_o}^\top \mathbf{v}_{w_c})}{\sum_{w \in V} \exp(\mathbf{u}_w^\top \mathbf{v}_{w_c})}$$

- Trong mô hình Skip-Gram của Word2Vec, xác suất dự đoán từ phụ thuộc vào tích vô hướng giữa hai vector từ, mà tích vô hướng lại phụ thuộc trực tiếp vào độ tương tự cosine.
- Khi các từ có nghĩa gần nhau, chúng thường xuất hiện trong những ngữ cảnh giống nhau, khiến các vector của chúng hướng gần nhau trong không gian \rightarrow độ tương tự cosine cao.

5. Demo



Vietnam National University Ho Chi Minh City
Ho Chi Minh University of Technology

WORD EMBEDDING

Mathematics Foundations for Computer Science

Many Thanks!

Ho Chi Minh City, May - 2025