Mô hình Word Embedding: Skip-gram, CBOW và GloVe

March 2025

Mục lục

Danh sách hình vẽ

Danh sách bảng

Chương 1

Embedding Từ (word2vec)

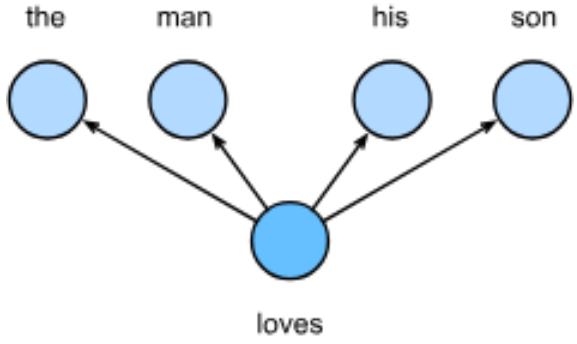
1.1 Mô hình Skip-Gram

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. Ví dụ, giả sử chuỗi văn bản là "the", "man", "loves", "his" và "son". Ta sử dụng "loves" làm từ đích trung tâm và đặt kích thước cửa số ngữ cảnh bằng 2. Như mô tả trong Fig. 14.1.1, với từ đích trung tâm "loves", mô hình skip-gram quan tâm đến xác suất có điều kiện sinh ra các từ ngữ cảnh ("the", "man", "his" và "son") nằm trong khoảng cách không quá 2 từ:

$$P(\text{"}the\text{"},\text{"}man\text{"},\text{"}his\text{"},\text{"}son\text{"} \mid \text{"}loves\text{"}).$$

Ta giả định rằng, với từ đích trung tâm cho trước, các từ ngữ cảnh được sinh ra độc lập với nhau. Trong trường hợp này, công thức trên có thể được viết lại thành

$$P(\text{"}the\text{"} \mid \text{"}loves\text{"}) \cdot P(\text{"}man\text{"} \mid \text{"}loves\text{"}) \cdot P(\text{"}his\text{"} \mid \text{"}loves\text{"}) \cdot P(\text{"}son\text{"} \mid \text{"}loves\text{"}).$$



Hình 1.1: Mô hình skip-gram quan tâm đến xác suất có điều kiện sinh ra các từ ngữ cảnh với một từ đích trung tâm cho trước.

Trong mô hình skip-gram, mỗi từ được biểu diễn bằng hai vector d-chiều để tính xác suất có điều kiện. Giả sử chỉ số của một từ trong từ điển là i, vector của từ được biểu diễn là $v_i \in \mathbb{R}^d$ khi từ này là từ đích trung tâm và là $u_i \in \mathbb{R}^d$ khi từ này là một từ ngữ cảnh. Gọi c và o lần lượt là chỉ số của từ đích trung tâm w_c và từ ngữ

cảnh w_o trong từ điển. Có thể thu được xác suất có điều kiện sinh ra từ ngữ cảnh cho một từ đích trung tâm cho trước bằng phép toán softmax trên tích vô hướng của vector:

$$P(w_o \mid w_c) = \frac{\exp(u_o^T v_c)}{\sum_{i \in \mathcal{V}} \exp(u_i^T v_c)},$$

trong đó, tập chỉ số trong bộ từ vựng là $\mathcal{V} = \{0, 1, \ldots, |\mathcal{V}| - 1\}$. Giả sử trong một chuỗi văn bản có độ dài T, từ tại bước thời gian t được ký hiệu là $w^{(t)}$. Giả sử rằng các từ ngữ cảnh được sinh độc lập với từ trung tâm cho trước. Khi kích thước cửa số ngữ cảnh là m, hàm hợp lý (likelihood) của mô hình skip-gram là xác suất kết hợp sinh ra tất cả các từ ngữ cảnh với bất kỳ từ trung tâm cho trước nào

$$\prod_{t=1}^{T} \prod_{-m \le j \le m, j \ne 0} P(w^{(t+j)} \mid w^{(t)}),$$

Ở đây, bất kỳ bước thời gian nào nhỏ hơn 1 hoặc lớn hơn T đều có thể được bỏ qua.

1.2 Huấn luyện Mô hình Skip-Gram

Các tham số trong mô hình skip-gram là vector từ đích trung tâm và vector từ ngữ cảnh cho từng từ riêng lẻ. Trong quá trình huấn luyện, chúng ta sẽ học các tham số mô hình bằng cách cực đại hóa hàm hợp lý, còn gọi là ước lượng hợp lý cực đại. Việc này tương tự với việc giảm thiểu hàm mất mát sau đây:

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

Nếu ta dùng SGD (Stochastic Gradient Descent), thì trong mỗi vòng lặp, ta chọn ra một chuỗi con nhỏ hơn bằng việc lấy mẫu ngẫu nhiên để tính toán mất mát cho chuỗi con đó, rồi sau đó tính gradient để cập nhật các tham số mô hình. Điểm then chốt của việc tính toán gradient là tính gradient của logarit xác suất có điều kiện cho vector từ trung tâm và vector từ ngữ cảnh. Đầu tiên, theo định nghĩa ta có

$$\log P(w_o | w_c) = u_o^T v_c - \log \left(\sum_{j \in \mathcal{V}} \exp(u_j^T v_c) \right).$$

Thông qua phép tính đạo hàm, ta nhận được giá trị gradient v_c từ công thức trên.

$$\frac{\partial \log P(w_o|w_c)}{\partial v_c} = u_o - \frac{\sum_{j \in \mathcal{V}} \exp(u_j^T v_c) u_j}{\sum_{i \in \mathcal{V}} \exp(u_i^T v_c)} = u_o - \sum_{j \in \mathcal{V}} \left(\frac{\exp(u_j^T v_c)}{\sum_{i \in \mathcal{V}} \exp(u_i^T v_c)} \right) u_j = u_o - \sum_{j \in \mathcal{V}} P(w_j|w_c) u_j.$$

Phép tính cho ra xác suất có điều kiện cho mọi từ có trong từ điển với từ đích trung tâm w_c cho trước. Sau đó, ta lại sử dụng phương pháp đó để tìm gradient cho các vector từ khác.

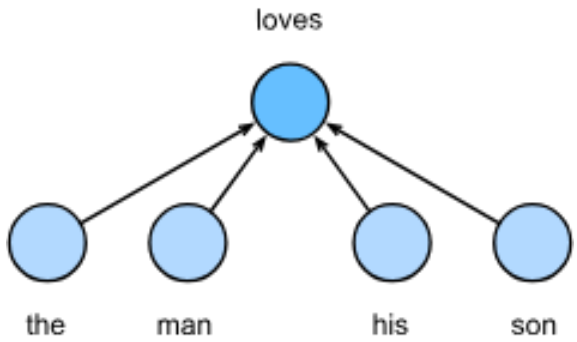
Sau khi huấn luyện xong, với từ bất kỳ có chỉ số là i trong từ điển, ta sẽ nhận được tập hai vector từ v_i và u_i. Trong các ứng dụng xử lý ngôn ngữ tự nhiên, vector từ đích trung tâm trong mô hình skip-gram thường được sử dụng để làm vector biểu diễn một từ.

1.3 Mô hình Túi từ Liên tục (CBOW)

Mô hình túi từ liên tục (*Continuous bag of words - CBOW*) tương tự như mô hình skip-gram. Khác biệt lớn nhất là mô hình CBOW giả định rằng từ đích trung tâm được tạo ra dựa trên các từ ngữ cảnh phía trước và sau nó trong một chuỗi văn bản. Với cùng một chuỗi văn bản gồm các từ "the", "man", "loves", "his" và "son", trong đó "loves" là từ đích trung tâm, với kích thước cửa số ngữ cảnh bằng 2, mô hình CBOW quan tâm đến xác suất có điều kiện để sinh ra từ đích "loves" dựa trên các từ ngữ cảnh "the", "man", "his" và "son" (minh họa ở Fig. 14.1.2) như sau:

$$P(\text{"loves"} \mid \text{"the"}, \text{"man"}, \text{"his"}, \text{"son"}).$$

Hình 1.2: Mô hình CBOW quan tâm đến xác suất có điều kiện tạo ra từ đích trung tâm dựa trên các từ ngữ cảnh cho trước.



Hình 1.3: Enter Caption

Vì có quá nhiều từ ngữ cảnh trong mô hình CBOW, ta sẽ lấy trung bình các vector từ của chúng và sau đó sử dụng phương pháp tương tự như trong mô hình skip-gram để tính xác suất có điều kiện. Giả sử $\mathbf{v}_i \in \mathbb{R}^d$ và $\mathbf{u}_i \in \mathbb{R}^d$ là vector từ ngữ cảnh và vector từ đích trung tâm của từ có chỉ số i trong từ điển (lưu ý rằng các ký hiệu này ngược với các ký hiệu trong mô hình skip-gram). Gọi c là chỉ số của từ đích trung tâm w_c, và $o_1, \ldots, o_{2m}$ là chỉ số các từ ngữ cảnh $w_{o_1}, \ldots, w_{o_{2m}}$ trong từ điển. Do đó, xác suất có điều kiện sinh ra từ đích trung tâm dựa vào các từ ngữ cảnh cho trước là

$$P(w_c \mid w_{o_1}, \ldots, w_{o_{2m}}) = \frac{\exp\left(\frac{1}{2m}\mathbf{u}_c^T(\mathbf{v}_{o_1} + \cdots + \mathbf{v}_{o_{2m}})\right)}{\sum_{i \in \mathcal{V}} \exp\left(\frac{1}{2m}\mathbf{u}_i^T(\mathbf{v}_{o_1} + \cdots + \mathbf{v}_{o_{2m}})\right)}.$$

Để rút gọn, ký hiệu $\mathcal{W}_o = \{w_{o_1}, \ldots, w_{o_{2m}}\}$, và $\bar{\mathbf{v}}_o = (\mathbf{v}_{o_1} + \cdots + \mathbf{v}_{o_{2m}})/(2m)$. Phương trình trên được đơn giản hóa thành

$$P(w_c \mid \mathcal{W}_o) = \frac{\exp(\mathbf{u}_c^T \bar{\mathbf{v}}_o)}{\sum_{i \in \mathcal{V}} \exp(\mathbf{u}_i^T \bar{\mathbf{v}}_o)}.$$

Cho một chuỗi văn bản có độ dài T, ta giả định rằng từ xuất hiện tại bước thời gian t là $w^{(t)}$, và kích thước của cửa số ngữ cảnh là m. Hàm hợp lý của mô hình CBOW là xác suất sinh ra bất kỳ từ đích trung tâm nào dựa vào những từ ngữ cảnh.

$$\prod_{t=1}^{T} P(w^{(t)} \mid w^{(t-m)}, \ldots, w^{(t-1)}, w^{(t+1)}, \ldots, w^{(t+m)}).$$

1.4 Huấn luyện Mô hình CBOW

Quá trình huấn luyện mô hình CBOW khá giống với quá trình huấn luyện mô hình skip-gram. Ước lượng hợp lý cực đại của mô hình CBOW tương đương với việc cực tiểu hóa hàm mất mát:

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

Lưu ý rằng

$$\log P(w_c | \mathcal{W}_o) = \mathbf{u}_c^T \bar{\mathbf{v}}_o - \log\left(\sum_{i \in \mathcal{V}} \exp(\mathbf{u}_i^T \bar{\mathbf{v}}_o)\right). \tag{14.1.14}$$

Thông qua phép đạo hàm, ta có thể tính log của xác suất có điều kiện của gradient của bất kỳ vector từ ngữ cảnh nào $\mathbf{v}_{o_i}$ $(i = 1, \ldots, 2m)$ trong công thức trên.

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

Sau đó, ta sử dụng cùng phương pháp đó để tính gradient cho các vector của từ khác. Không giống như mô hình skip-gram, trong mô hình CBOW ta thường sử dụng vector từ ngữ cảnh làm vector biểu diễn một từ.

Chương 2

Mô hình GloVe

2.1 Embedding từ với Vector Toàn cục (GloVe)

Trước tiên, ta sẽ xem lại mô hình skip-gram trong word2vec. Xác suất có điều kiện $P(w_j \mid w_i)$ được biểu diễn trong mô hình skip-gram bằng hàm kích hoạt softmax sẽ được gọi là q_{ij} như sau:

$$q_{ij} = \frac{\exp(\mathbf{u}_j^T \mathbf{v}_i)}{\sum_{k \in \mathcal{V}} \exp(\mathbf{u}_k^T \mathbf{v}_i)},$$

(14.5.1)

Ở đây $\mathbf{v}_i$ và $\mathbf{u}_i$ là các biểu diễn vector từ w_i với chỉ số i, lần lượt khi nó là từ trung tâm và từ ngữ cảnh, và $\mathcal{V} = \{0, 1, \ldots, |\mathcal{V}| - 1\}$ là tập chứa các chỉ số của bộ từ vựng.

Từ w_i có thể xuất hiện trong tập dữ liệu nhiều lần. Ta gom tất cả các từ ngữ cảnh mỗi khi w_i là từ trung tâm và giữ các lần trùng lặp, rồi ký hiệu đó là tập bội C_i. Số lượng của một phần tử trong tập bội được gọi là bội số của phần tử đó. Chẳng hạn, giả sử rằng từ w_i xuất hiện hai lần trong tập dữ liệu: khi hai từ w_i đó là từ trung tâm trong chuỗi văn bản, hai cửa sổ ngữ cảnh tương ứng chứa các chỉ số từ ngữ cảnh 2, 1, 5, 2 và 2, 3, 2, 1. Khi đó, ta sẽ có tập bội $C_i = \{1, 1, 2, 2, 2, 2, 3, 5\}$, trong đó bội số của phần tử 1 là 2, bội số của phần tử 2 là 4, và bội số của phần tử 3 và 5 đều là 1. Ta ký hiệu bội số của phần tử j trong tập bội C_i là x_{ij}: nó là số lần từ w_j xuất hiện trong cửa sổ ngữ cảnh khi từ trung tâm là w_i trong toàn bộ tập dữ liệu. Kết quả là hàm mất mát của mô hình skip-gram có thể được biểu diễn theo một cách khác:

$$-\sum_{i \in \mathcal{V}} \sum_{j \in \mathcal{V}} x_{ij} \log q_{ij}.$$

(14.5.2)

Ta tính tổng số lượng tất cả các từ ngữ cảnh đối với từ trung tâm w_i để có x_i, rồi thu được xác suất có điều kiện để sinh ra từ ngữ cảnh w_j dựa trên từ trung tâm w_i là p_{ij} bằng x_{ij}/x_i. Ta có thể viết lại hàm mất mát của mô hình skip-gram như sau

$$-\sum_{i\in\mathcal{V}} x_i \sum_{j\in\mathcal{V}} p_{ij}\log q_{ij}.$$

(14.5.3)

Trong công thức trên, $\sum_{j\in\mathcal{V}} p_{ij}\log q_{ij}$ tính toán phân phối xác suất có điều kiện p_{ij} của việc sinh từ ngữ cảnh dựa trên từ đích trung tâm w_i và entropy chéo với phân phối xác suất có điều kiện q_{ij} được dự đoán bởi mô hình. Hàm mất mát được đánh trọng số bằng cách sử dụng tổng số từ ngữ cảnh cho từ đích trung tâm w_i. Việc cực tiểu hóa hàm mất mát theo công thức trên cho phép phân phối xác suất có điều kiện được dự đoán một cách gần nhất có thể tới phân phối xác suất có điều kiện thật sự.

Tuy nhiên, mặc dù là hàm mất mát phổ biến nhất, đôi khi hàm mất mát entropy chéo lại không phải là một lựa chọn phù hợp. Một mặt, như ta đã đề cập trong Section 14.2, chi phí để mô hình đưa ra dự đoán q_{ij} trở thành phân phối xác suất hợp lệ gồm phép lấy tổng qua toàn bộ các từ trong từ điển ở mẫu số của nó. Điều này có thể dễ dàng khiến tổng chi phí tính toán trở nên quá lớn. Mặt khác, thường sẽ có rất nhiều từ hiếm gặp trong từ điển, và chúng hiếm khi xuất hiện trong tập dữ liệu. Trong hàm mất mát entropy chéo, dự đoán cuối cùng cho phân phối xác suất có điều kiện trên một lượng lớn các từ hiếm gặp rất có thể sẽ không được chính xác.

2.2 Mô hình GloVe

Để giải quyết vấn đề trên, GloVe [**?**], một mô hình embedding từ xuất hiện sau word2vec đã áp dụng mất mát bình phương và đề xuất ba thay đổi trong mô hình skip-gram dựa theo mất mát này.

1. Ở đây, ta sử dụng các biến phân phối phi xác suất $p'_{ij} = x_{ij}$ và $q'_{ij} = \exp(\mathbf{u}_j^T\mathbf{v}_i)$ rồi tính log của chúng. Do đó, ta có mất mát bình phương:

$$(\log p'_{ij} - \log q'_{ij})^2 = (\mathbf{u}_j^T\mathbf{v}_i - \log x_{ij})^2.$$

2. Ta thêm hai tham số mô hình cho mỗi từ w_i: hệ số điều chỉnh b_i (cho các từ trung tâm) và c_j (cho các từ ngữ cảnh).

3. Thay thế trọng số của mỗi giá trị mất mát bằng hàm $h(x_{ij})$. Hàm trọng số $h(x)$ là hàm đơn điệu tăng trong khoảng $[0, 1]$.

Do đó, mục tiêu của GloVe là cực tiểu hóa hàm mất mát:

$$\sum_{i\in\mathcal{V}}\sum_{j\in\mathcal{V}} h(x_{ij})(\mathbf{u}_j^T\mathbf{v}_i + b_i + c_j - \log x_{ij})^2.$$

(14.5.4)

Ở đây, chúng tôi có một đề xuất đối với việc lựa chọn hàm trọng số $h(x)$: khi $x < c$ (ví dụ $c = 100$) thì $h(x) = (x/c)^\alpha$ (ví dụ $\alpha = 0.75$), nếu không thì $h(x) = 1$. Do $h(0) = 0$, ta có thể đơn thuần bỏ qua mất mát bình phương tại $x_{ij} = 0$. Khi sử

dụng minibatch SGD trong quá trình huấn luyện, ta tiến hành lấy mẫu ngẫu nhiên để được một minibatch x_{ij} khác không tại mỗi bước thời gian và tính toán gradient để cập nhật các tham số mô hình. Các giá trị x_{ij} khác không trên được tính trước trên toàn bộ tập dữ liệu và là thống kê toàn cục của tập dữ liệu. Do đó, tên gọi GloVe được lấy từ "*Global Vectors*" (Vector Toàn cục).

Chú ý rằng nếu từ w_i xuất hiện trong cửa sổ ngữ cảnh của từ w_j thì từ w_j cũng sẽ xuất hiện trong cửa sổ ngữ cảnh của từ w_i. Do đó, $x_{ij} = x_{ji}$. Không như word2vec, GloVe khớp $\log x_{ij}$ đối xứng thay vì xác suất có điều kiện p_{ij} bất đối xứng. Do đó, vector từ đích trung tâm và vector từ ngữ cảnh của bất kì từ nào đều tương đương nhau trong GloVe. Tuy vậy, hai tập vector từ được học bởi cùng một mô hình về cuối có thể sẽ khác nhau do giá trị khởi tạo khác nhau. Sau khi học tất cả các vector từ, GloVe sẽ sử dụng tổng của vector từ đích trung tâm và vector từ ngữ cảnh để làm vector từ cuối cùng cho từ đó.

2.3 Lý giải GloVe bằng Tỷ số Xác suất Có điều kiện

Ta cũng có thể cố gắng lý giải embedding từ GloVe theo một cách nhìn khác. Ta sẽ tiếp tục sử dụng các ký hiệu như ở trên, $P(w_j|w_i)$ biểu diễn xác suất có điều kiện sinh từ ngữ cảnh w_j với từ tâm đích w_i trong tập dữ liệu, và xác suất này được ghi lại bằng p_{ij}. Xét ví dụ thực tế từ một kho ngữ liệu lớn, ở đây ta có hai tập các xác suất có điều kiện với "ice" và "steam" là các từ tâm đích và tỷ số giữa chúng:

Bảng 2.1: Tỷ số xác suất có điều kiện

$w_k =$	solid	gas	water	fashion
$p_1 = P(w_k\|\text{ice})$	0.00019	0.000066	0.003	0.000017
$p_2 = P(w_k\|\text{steam})$	0.000022	0.00078	0.0022	0.000018
p_1/p_2	8.9	0.085	1.36	0.96

Ta có thể quan sát thấy các hiện tượng như sau:

- Với từ w_k liên quan tới từ "ice" (đá) nhưng không liên quan đến từ "steam" (hơi nước), như là w_k = solid (rắn), ta kỳ vọng tỷ số xác suất có điều kiện sẽ lớn hơn (8.9).
- Với từ w_k liên quan tới từ "steam" mà không liên quan tới "ice", như w_k = gas (khí), ta kỳ vọng tỷ số sẽ nhỏ hơn (0.085).
- Với từ w_k liên quan tới cả hai như water (nước), tỷ số gần 1 (1.36).
- Với từ không liên quan như fashion (thời trang), tỷ số cũng gần 1 (0.96).

Có thể thấy rằng tỷ số xác suất có điều kiện thể hiện quan hệ giữa các từ trực quan hơn. Ta có thể xây dựng hàm vector để khớp tỷ số này:

$$f(\mathbf{u}_j, \mathbf{u}_k, \mathbf{v}_i) \approx \frac{p_{ij}}{p_{ik}}.$$

(14.5.5)

Chọn hàm f thỏa mãn $f(x)f(-x) = 1$, ta có thể dùng $f(x) = \exp(x)$:

$$f(\mathbf{u}_j, \mathbf{u}_k, \mathbf{v}_i) = \frac{\exp(\mathbf{u}_j^T \mathbf{v}_i)}{\exp(\mathbf{u}_k^T \mathbf{v}_i)} \approx \frac{p_{ij}}{p_{ik}}.$$

(14.5.6)

Giả sử $\exp(\mathbf{u}_j^T \mathbf{v}_i) \approx \alpha p_{ij}$ với α là hằng số, sau khi lấy logarit và thêm hệ số điều chỉnh b_i (từ trung tâm) và c_j (từ ngữ cảnh), ta có:

$$\mathbf{u}_j^T \mathbf{v}_i + b_i + c_j \approx \log(x_{ij}).$$

(14.5.7)

Hàm mất mát GloVe được xây dựng từ sai số bình phương của biểu thức này.

2.4 So sánh với Word2Vec

Bảng 2.2: So sánh GloVe và Word2Vec

Đặc điểm	Word2Vec (Skip-gram)	GloVe
Dữ liệu đầu vào	Cửa sổ ngữ cảnh cục bộ	Ma trận đồng xuất hiện toàn cục
Tính đối xứng	Không đối xứng	Đối xứng ($x_{ij} = x_{ji}$)
Xử lý từ hiếm	Kém hiệu quả	Tốt hơn nhờ hàm trọng số
Tốc độ	Nhanh với Negative Sampling	Chậm hơn do tính toán ma trận

2.5 Kết luận

GloVe cung cấp cách tiếp cận hiệu quả để học embedding từ bằng cách:

- Kết hợp thông tin toàn cục từ ma trận đồng xuất hiện
- Tối ưu hóa trực tiếp để bảo toàn quan hệ giữa các từ
- Đặc biệt hiệu quả với các từ hiếm nhờ cơ chế trọng số

Tài liệu tham khảo

[1] Aston Zhang, Zack C. Lipton, Mu Li, & Alex J. Smola. **Dive into Deep Learning**. URL: `https://d2l.ai`