TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI
VIỆN CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG

———— * ————

BÁO CÁO
BÀI TẬP LỚN MÔN HỌC
XỬ LÝ NGÔN NGỮ TỰ NHIÊN

Tên đề tài:

Speech and Language Processing

Giảng viên hướng dẫn: **TS. Nguyễn Kiêm Hiếu**

Học viên thực hiện: **Trịnh Trường Giang**

Mã học viên: CB190205

HÀ NỘI 05-2020

TÓM TẮT BÁO CÁO

Các công việc thực hiện:

	Công việc thực hiện
Tuần 1	- Đọc, tìm hiểu chương 15
Tuần 2	- Đọc, tìm hiểu chương 16
Tuần 3	- Đọc, tìm hiểu chương 17
Tuần 4	- Đọc, tìm hiểu chương 18

MỤC LỤC

DANH MỤC CÁC HÌNH ẢNH, BẢNG BIỂU TRONG BÁO CÁO

1. Chương 15: Vector Semantics

Trong chương này sẽ giới thiệu các phương pháp sử dụng phân phối, trong đó ý nghĩa của một từ (word) được tính qua việc phân phối các từ xung quanh nó. Những từ này thường được biểu diễn dưới dạng một vectơ hoặc mảng các số liên quan theo một cách nào đó để đếm, và vì vậy các phương thức này thường được gọi là ngữ nghĩa vectơ (Vector Semantics).

Trong chương này, giới thiệu một phương pháp đơn giản trong đó nghĩa của từ được định nghĩa đơn giản bằng tần suất xuất hiện gần các từ khác. Chúng ta sẽ thấy rằng phương pháp này dẫn đến các vectơ rất dài (về mặt kỹ thuật, còn gọi là: high dimensional), rất thưa thớt, tức là chứa hầu hết các số không (vì rất nhiều các từ đơn giản không bao giờ xảy ra trong ngữ cảnh của các từ khác).

1.1: Words và Vectors

Vectơ hoặc các mô hình phân phối ý nghĩa thường dựa trên ma trận đồng xuất hiện (co-occurrence), một cách biểu thị mức độ thường xuyên xảy ra cùng nhau của các từ. Một ma trận đồng xuất hiện ví dụ như một ma trận term-document matrix.

1.1.1: Vectors and documents.

term-document matrix

Trong một term-document matrix, mỗi hàng đại diện cho một từ trong từ vựng và mỗi cột đại diện cho một tài liệu từ một số bộ sưu tập. *Hình 1:* cho thấy một phần nhỏ từ một term-document matrix cho thấy sự xuất hiện của bốn từ trong bốn vở kịch của Shakespeare. Mỗi ô trong ma trận này biểu thị số lần một từ cụ thể (được xác định bởi hàng) xảy ra trong một tài liệu cụ thể (được xác định bởi cột). Do đó “clown” xuất hiện 117 lần trong “Twelfth Night”.

	As You Like It	Twelfth Night	Julius Caesar	Henry V
battle	1	1	8	15
soldier	2	2	12	36
fool	37	58	1	5
clown	5	117	0	0

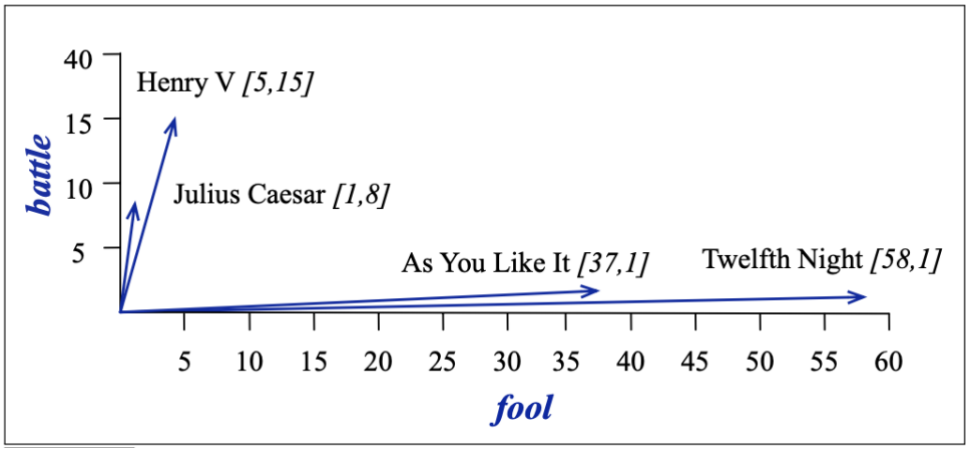
Hình 1:Term-document matrix.

Term-document matrix của *Hình 1* lần đầu tiên được định nghĩa là một phần của mô hình không gian vectơ (vector space model) của truy xuất thông tin (Salton, 1971). Trong mô hình này, một tài liệu được biểu diễn dưới dạng vectơ đếm, là một cột trong *Hình 1*.

vector space

Một không gian vectơ là một tập hợp các vectơ, được đặc trưng bởi số chiều của chúng. Thứ tự của các chiều trong một không gian vectơ không phải là tùy ý; mỗi vị trí chỉ ra một chiều có ý nghĩa mà ở đó các tài liệu có thể khác nhau. Do đó, chiều đầu tiên cho cả hai vectơ “As You Like It” và “Twelfth Night” tương ứng với số lần xuất hiện từ “battle” và chúng ta có thể so sánh từng chiều, ví dụ như các vectơ cho “As You Like It” và “Twelfth Night” có cùng giá trị 1 cho chiều đầu tiên. Chúng ta có thể nghĩ về vectơ cho một tài liệu như xác định một

điểm trong không gian V chiều; do đó, các tài liệu trong *Hình 1* là các điểm trong không gian 4 chiều. Vì không gian 4 chiều khó vẽ và tưởng tượng, *Hình 2*: cho thấy một hình ảnh trực quan theo hai chiều.



Hình 2: Trực quan của các document vector.

Term-document matrix ban đầu được định nghĩa là một phương tiện để tìm các tài liệu tương tự cho nhiệm vụ truy xuất thông tin tài liệu (document information retrieval). Hai tài liệu tương tự nhau sẽ có xu hướng có từ tương tự nhau và nếu hai tài liệu có từ tương tự thì vectơ cột của chúng sẽ có xu hướng tương tự nhau. Các vectơ cho các phim hài "As You Like It" [1,2,37,5] và "Twelfth Night" [1,2,58,117] trông rất giống nhau (nhiều fool và clown hơn soldier và battle) so với "Julius Caesar" [8,12,1,0] hoặc "Henry V" [15,36,5,0]. Chúng ta có thể thấy trực giác với các số nguyên; ở chiều thứ nhất (trận chiến), các vở hài kịch có số lượng thấp và những cái khác có số lượng cao, và chúng ta có thể thấy nó một cách trực quan trong *Hình 2*; chúng ta sẽ sớm thấy cách định lượng trực giác này một cách chính thức hơn. Tất nhiên, một Term-document matrix thực sự sẽ không có 4 hàng và cột, chứ đừng nói 2. Nói chung, Term-document matrix X có |V| hàng (số từ trong tập từ vựng) và D cột (một cho mỗi tài liệu trong bộ sưu tập); Ta sẽ thấy, kích thước từ vựng thường ít nhất là hàng chục nghìn và số lượng tài liệu có thể rất lớn (hãy nghĩ về tất cả các trang trên web).

1.1.2: Words as vectors.

Chúng ta đã thấy rằng các tài liệu có thể được biểu diễn dưới dạng vectơ trong không gian vectơ. Nhưng Vector Semantics cũng có thể được sử dụng để thể hiện ý nghĩa của các từ (words), bằng cách liên kết mỗi từ với một vectơ.

row vector

Vectơ từ (Word Vector) bây giờ là một vectơ hàng chứ không phải là vectơ cột và do đó kích thước của vectơ đã khác. Bốn chiều của vectơ cho "fool" - [37,58,1,5], tương ứng với bốn vở kịch của Shakespeare. Tương tự, được sử dụng để tạo thành các vectơ cho 3 từ còn lại: "clown" - [5, 117, 0, 0]; "battle" - [1,1,8,15]; và "soldier" - [2,2,12,36]. Do đó, mỗi mục trong vectơ đại diện cho số lượng từ xuất hiện trong một tài liệu tương ứng với chiều đó. Đối với các tài liệu, chúng tôi thấy rằng các tài liệu tương tự có vectơ tương tự, bởi vì các tài liệu tương tự có xu hướng có các từ tương tự. Nguyên tắc tương tự này áp dụng cho các từ: các từ tương tự có vectơ tương tự vì chúng có xu hướng xảy ra trong các tài liệu tương tự. Do đó, Term-document matrix cho phép chúng ta biểu thị nghĩa của một từ bằng các tài liệu mà nó có xu hướng xảy ra.

term-term matrix, word-word matrix

Tuy nhiên, thông thường nhất là sử dụng một loại bối cảnh khác cho các chiều của biểu diễn vectơ từ. Thay vì Term-document matrix, chúng ta sử dụng **term-term matrix**, thường được gọi là ma trận từ-từ (**word-word matrix**) hoặc **term-context matrix**, trong đó các cột được gắn nhãn bằng các từ thay vì các tài liệu. Do đó, ma trận này có kích thước |V| × |V| và mỗi ô ghi lại số lần từ ở hàng (mục tiêu) và từ ở cột (ngữ cảnh) cùng xuất hiện trong một số ngữ cảnh trong các tài liệu tập huấn luyện. Bối cảnh (**context**) có thể là tài liệu, trong trường hợp đó, ô biểu thị số lần hai từ xuất hiện trong cùng một tài liệu. Tuy nhiên, phổ biến nhất là sử dụng các bối cảnh nhỏ hơn, nói chung là một cửa sổ (window) xung quanh từ, ví dụ 4 từ bên trái và 4 từ bên phải, trong trường hợp đó, ô biểu thị số lần (trong corpus huấn luyện) từ ở cột xảy ra trong một cửa sổ từ ± 4 xung quanh từ ở hàng.

	aardvark	...	computer	data	pinch	result	sugar	...
apricot	0	...	0	0	1	0	1	
pineapple	0	...	0	0	1	0	1	
digital	0	...	2	1	0	1	0	
information	0	...	1	6	0	4	0	

Hình 3: (Word vectors) Vector từ.

Lưu ý rằng |V|, độ dài của vectơ, thường là kích thước của từ bộ vựng, thường là từ 10.000 đến 50.000 từ (sử dụng các từ thường xuyên nhất trong tập huấn luyện; giữ các từ sau khoảng 50.000 thường xuyên nhất hoặc thường không hữu ích) . Nhưng tất nhiên vì hầu hết các số này đều bằng 0 nên đây là các biểu diễn vectơ thưa thớt và có các thuật toán hiệu quả để lưu trữ và tính toán với ma trận thưa thớt. Kích thước của cửa sổ được sử dụng để thu thập số lượng có thể thay đổi dựa trên các mục tiêu của biểu diễn, nhưng thường nằm trong khoảng từ 1 đến 8 từ mỗi phía của từ mục tiêu (cho tổng số ngữ cảnh từ 3-17 từ). Nói chung, cửa sổ càng ngắn, các biểu diễn càng cú pháp (syntactic), vì thông tin đến từ các từ gần đó ngay lập tức; Cửa sổ càng dài, các mối quan hệ càng có ý nghĩa (semantic).

Chúng ta đã nói một cách cơ bản về tương đồng (similarity), nhưng thường hữu ích để phân biệt hai loại tương đồng hoặc liên kết giữa các từ (Schutze và Pedersen, 1993). Hai từ có **first-order co-occurrence** (đôi khi được gọi là liên kết cú pháp) nếu chúng thường ở gần nhau. Ví dụ, “viết” là một **first-order co-occurrence** của “cuốn sách” hoặc “bài thơ”. Hai từ có **second-order co-occurrence** (đôi khi được gọi là liên kết nghịch lý) nếu chúng có hàng xóm tương tự. Ví dụ, “viết” là một **second-order co-occurrence** của các từ như “nói” hoặc “nhận xét”.

Bây giờ chúng ta đã có một số trực giác, hãy tiếp tục để kiểm tra các chi tiết tính toán một biểu diễn vectơ cho một từ. Chúng ta sẽ bắt đầu với một trong những biểu diễn vectơ được sử dụng phổ biến nhất: PPMI hoặc Positive Pointwise Mutual Information.

1.2: Weighing terms: Pointwise Mutual Information (PMI)

Ma trận đồng xuất hiện trong *Hình 3* thể hiện mỗi ô bằng tần số thô của sự đồng xuất hiện của hai từ. Tuy nhiên, hóa ra tần số đơn giản đó không phải là thước đo liên kết tốt nhất giữa các từ. Một vấn đề là tần số thô rất lệch và không phân biệt đối xử. Nếu chúng ta muốn biết những loại bối cảnh nào được chia sẻ bởi “apricot” và “pineapple” nhưng không phải bằng

"digital" và "information", chúng ta sẽ không nhận được sự phân biệt tốt từ các từ "the", "it", or "they", xảy ra thường xuyên với tất cả các loại từ và không có thông tin về bất kỳ từ cụ thể.

Thay vào đó, chúng ta thích các từ ngữ cảnh đặc biệt nhiều thông tin về từ mục tiêu. Trọng số hoặc thước đo liên kết tốt nhất giữa các từ nên cho chúng ta biết mức độ thường xuyên hơn là khả năng hai từ cùng xuất hiện.

Mutual information

Pointwise Mutual information là một biện pháp như vậy. Nó được đề xuất bởi Church and Hanks (1989) và (Church and Hanks, 1990), dựa trên khái niệm thông tin lẫn nhau (Mutual information). Mutual information giữa hai biến ngẫu nhiên X và Y là

$$I(X,Y) = \sum_x \sum_y P(x,y) \log_2 \frac{P(x,y)}{P(x)P(y)}$$

Pointwise Mutual information (Fano, 1961) là thước đo mức độ thường xuyên xảy ra hai sự kiện x và y, so với những gì chúng ta mong đợi nếu chúng độc lập:

$$I(x,y) = \log_2 \frac{P(x,y)}{P(x)P(y)}$$

Chúng ta có thể áp dụng trực giác này cho các vectơ đồng xuất hiện bằng cách xác định Pointwise Mutual information giữa một từ mục tiêu w và một từ ngữ cảnh c là:

$$\text{PMI}(w,c) = \log_2 \frac{P(w,c)}{P(w)P(c)}$$

Tử số cho chúng ta biết tần suất chúng ta quan sát hai từ với nhau (giả sử chúng ta tính xác suất bằng cách sử dụng MLE). Mẫu số cho chúng ta biết mức độ thường xuyên chúng ta mong đợi hai từ cùng xuất hiện giả sử chúng từng xảy ra độc lập, do đó xác suất của chúng có thể được nhân lên. Do đó, tỷ lệ này cho chúng ta ước tính số lượng mục tiêu và tính năng cùng xảy ra nhiều hơn chúng ta mong đợi. Giá trị PMI nằm trong khoảng từ âm đến dương vô cực. Nhưng các giá trị PMI âm (ngụ ý mọi thứ xảy ra ít thường xuyên hơn chúng ta mong đợi) có xu hướng không đáng tin trừ khi corpora của chúng ta rất lớn.

Để phân biệt xem hai từ có xác suất riêng lẻ của mỗi từ = 10 ^ -6 xảy ra với nhau thường xuyên hơn cơ hội hay không, chúng ta cần chắc chắn rằng xác suất của hai từ đó xảy ra khác nhau đáng kể so với 10 ^ -12, và loại độ chi tiết này sẽ đòi hỏi một khối lượng lớn corpus. Hơn nữa, không rõ liệu thậm chí có thể đánh giá điểm "không liên quan" như vậy với các đánh giá của con người hay không. Vì lý do này, người ta thường sử dụng Positive PMI (được gọi là PPMI) thay thế tất cả các giá trị PMI âm bằng 0 (Church và Hanks 1989, Dagan et al. 1993, Niwa và Nitta 1994):

$$\text{PPMI}(w,c) = \max(\log_2 \frac{P(w,c)}{P(w)P(c)}, 0)$$

Chính thức hơn, giả sử chúng ta có ma trận đồng xuất hiện F với các hàng W (từ) và cột C (bối cảnh), trong đó f_{ij} đưa ra số lần từ w_i xảy ra trong ngữ cảnh c_i. Điều này có thể được biến thành ma trận PPMI trong đó $PPMI_{ij}$ đưa ra giá trị PPMI của từ w_i với ngữ cảnh c_j như sau:

$$p_{ij} = \frac{f_{ij}}{\sum_{i=1}^{W}\sum_{j=1}^{C} f_{ij}} \quad p_{i*} = \frac{\sum_{j=1}^{C} f_{ij}}{\sum_{i=1}^{W}\sum_{j=1}^{C} f_{ij}} \quad p_{*j} = \frac{\sum_{i=1}^{W} f_{ij}}{\sum_{i=1}^{W}\sum_{j=1}^{C} f_{ij}}$$

$$\text{PPMI}_{ij} = \max(\log_2 \frac{p_{ij}}{p_{i*}p_{*j}}, 0)$$

Do đó, ví dụ, chúng ta có thể tính toán PPMI (w = information, c = data), giả sử chúng ta coi rằng *Hình 3* bao gồm tất cả các bối cảnh liên quan:

$$P(\text{w=information,c=data}) = \frac{6}{19} = .316$$
$$P(\text{w=information}) = \frac{11}{19} = .579$$
$$P(\text{c=data}) = \frac{7}{19} = .368$$
$$\text{ppmi(information,data)} = \log 2(.316/(.368 * .579)) = .568$$

Hình 4 cho thấy các xác suất chung được tính toán từ các số liệu trong *Hình 3,* và *Hình 5* cho thấy các giá trị PPMI.

	p(w,context)					p(w)
	computer	**data**	**pinch**	**result**	**sugar**	**p(w)**
apricot	0	0	0.05	0	0.05	0.11
pineapple	0	0	0.05	0	0.05	0.11
digital	0.11	0.05	0	0.05	0	0.21
information	0.05	.32	0	0.21	0	0.58
p(context)	0.16	0.37	0.11	0.26	0.11	

Hình 4: Biến đổi từ tần số thành xác suất trong ma trận đồng xuất hiện.

	computer	**data**	**pinch**	**result**	**sugar**
apricot	0	0	2.25	0	2.25
pineapple	0	0	2.25	0	2.25
digital	1.66	0	0	0	0
information	0	0.57	0	0.47	0

Hình 5: Biến đổi các giá trị trong ma trận đồng xuất hiện thành giá trị PPMI.

PMI có vấn đề thiên vị (bias) đối với các sự kiện không thường xuyên; những từ "rất hiếm" có xu hướng có giá trị PMI rất cao. Một cách để giảm sự thiên vị này đối với các sự kiện tần số thấp là thay đổi một chút tính toán cho P(c), sử dụng hàm khác Pα (c) (Levy et al., 2015):

$$\text{PPMI}_\alpha(w,c) = \max(\log_2 \frac{P(w,c)}{P(w)P_\alpha(c)}, 0)$$

$$P_\alpha(c) = \frac{count(c)^\alpha}{\sum_c count(c)^\alpha}$$

Levy và cộng sự. (2015) đã phát hiện ra rằng cài đặt α = 0,75 đã cải thiện hiệu suất của các nhúng trong một loạt các nhiệm vụ (dựa trên trọng số tương tự được sử dụng cho Skip-gram (Mikolov et al., 2013a) và GloVe (Pennington et al., 2014)). Điều này hoạt động vì việc tăng xác suất lên α = 0,75 làm tăng xác suất được gán cho các bối cảnh hiếm, và do đó làm giảm PMI của họ (Pα (c)> P (c) khi c hiếm).

Một giải pháp khả thi khác là Laplace smoothing: Trước khi tính toán PMI, một hằng số k nhỏ (giá trị 0,1-3 là phổ biến) được thêm vào mỗi số đếm, thu hẹp (chiết khấu) tất cả các giá trị khác không. K càng lớn, càng nhiều giá trị khác không được thu hẹp.

	computer	data	pinch	result	sugar
apricot	2	2	3	2	3
pineapple	2	2	3	2	3
digital	4	3	2	3	2
information	3	8	2	6	2

Hình 6: Laplace Smoothing (Thêm 2) vào ma trận Hình 3.

	computer	data	pinch	result	sugar
apricot	0	0	0.56	0	0.56
pineapple	0	0	0.56	0	0.56
digital	0.62	0	0	0	0
information	0	0.58	0	0.37	0

Hình 7: PPMI Matrix từ Hình 6.

1.2.1: Các lựa chọn thay thế cho PPMI để đo lường độ liên kết.

Mặc dù PPMI khá phổ biến, nhưng đây không phải là biện pháp đo độ liên kết duy nhất giữa hai từ. Các biện pháp đo độ liên kết phổ biến khác đến từ trích xuất thông tin như (**tf-idf**, **Dice**) hoặc từ kiểm tra giả thuyết (**t-test**, kiểm tra tỷ lệ khả năng – **likelihood-ratio tes**t). Trong phần này, chúng ta tóm tắt ngắn gọn một trong những loại biện pháp này.

Trước tiên, hãy xem xét sơ đồ trọng số chuẩn cho **Term-document matrix** trong trích xuất thông tin, được gọi là **Tf-idf**. Tf-idf (là dấu gạch nối, không phải dấu trừ) là sản phẩm của hai yếu tố. Đầu tiên là **Term frequency** (Luhn, 1957): đơn giản là tần số của các từ trong tài liệu, mặc dù chúng ta cũng có thể sử dụng các hàm của tần số này như hàm log.

Yếu tố thứ hai được sử dụng để mang lại trọng số cao hơn cho các từ chỉ xảy ra trong một vài tài liệu. Các term trong một vài tài liệu rất hữu ích để phân biệt các tài liệu đó với phần còn lại của bộ sưu tập; các term thường xuyên trên toàn bộ bộ sưu tập không hữu ích. Tần số tài liệu nghịch đảo (**inverse document frequency** – **IDF**) (Sparck Jones, 1972) là một cách gán trọng số cao hơn cho những từ phân biệt đối xử này. IDF được xác định bằng cách sử dụng phân số N / df_i, trong đó N là tổng số tài liệu trong bộ sưu tập và df_i là số lượng tài liệu trong đó thuật ngữ i xảy ra. Càng ít tài liệu trong đó một thuật ngữ xảy ra, trọng số này càng cao. Trọng số thấp nhất là 1 được gán cho các điều khoản xảy ra trong tất cả các tài liệu. Do số lượng lớn tài liệu trong nhiều bộ sưu tập, biện pháp này thường được nén với hàm log. Do đó, định nghĩa IDF là:

$$\text{idf}_i = \log\left(\frac{N}{df_i}\right)$$

Kết hợp **TF** với **IDF** dẫn đến một sơ đồ được gọi là trọng số **tf-idf** của giá trị cho từ i trong tài liệu j, w_{ij} :

$$w_{ij} = \text{tf}_{ij}\text{idf}_i$$

Do đó, Tf-idf thích các từ thường xuyên trong tài liệu hiện tại j, nhưng hiếm khi xuất hiện chung chung trong bộ sưu tập. Trọng số tf-idf cho đến nay là cách chiếm ưu thế của ma trận đồng xảy ra trong trích xuất thông tin, nhưng cũng đóng một vai trò trong nhiều khía cạnh khác của xử lý ngôn ngữ tự nhiên bao gồm cả tóm tắt. Tuy nhiên, Tf-idf thường không được sử dụng như một thành phần trong các biện pháp word similarity; khi đó PPMI và các số liệu kiểm tra ý nghĩa như t-test và likelihood-ratio là phổ biến hơn.

Thống kê t-test, như PMI, có thể được sử dụng để đo lường mức độ thường xuyên của liên quan hơn là cơ hội. Thống kê t-test tính toán sự khác biệt giữa các phương tiện được quan sát và dự kiến, được chuẩn hóa bởi phương sai. Giá trị của t càng cao, khả năng chúng ta có thể bác bỏ giả thuyết khống cho rằng các phương tiện được quan sát và mong đợi là như nhau.

$$t = \frac{\bar{x} - \mu}{\sqrt{\frac{s^2}{N}}}$$

Khi được áp dụng để liên kết giữa các từ, giả thuyết null là hai từ này độc lập và do đó P(a, b) = P(a) P(b) mô hình chính xác mối quan hệ giữa hai từ. Chúng tôi muốn biết khả năng P (a, b) thực tế của MLE khác với giá trị giả thuyết null này, được chuẩn hóa bằng phương sai. Phương sai s2 có thể được xấp xỉ bằng xác suất dự kiến P (a) P (b) (xem Manning và Schu¨tze (1999)). Bỏ qua N (vì nó là hằng số), do đó, phép đo liên kết t-test kết quả là (Curran, 2003):

$$\text{t-test}(a,b) = \frac{P(a,b) - P(a)P(b)}{\sqrt{P(a)P(b)}}$$

1.3: Measuring similarity (Đo độ tương tự): the cosine.

Để xác định độ tương tự giữa hai từ đích v và w, chúng ta cần một thước đo để lấy hai vectơ như vậy và đưa ra thước đo độ tương tự của vectơ. Cho đến nay, số liệu tương tự phổ biến nhất là cosin của góc giữa các vectơ. Trong phần này, chúng ta trình bày và giới thiệu biện pháp quan trọng này. Các cosine giống như hầu hết các biện pháp tương tự vectơ (vector similarity) được sử dụng trong NLP, dựa trên toán tử **dot product** từ đại số tuyến tính, còn được gọi là **Inner product**:

$$\text{dot-product}(\vec{v},\vec{w}) = \vec{v}\cdot\vec{w} = \sum_{i=1}^{N} v_i w_i = v_1 w_1 + v_2 w_2 + \ldots + v_N w_N$$

Như chúng ta sẽ thấy, hầu hết các số liệu về độ tương tự giữa các vectơ đều dựa trên **dot product**. **Dot product** hoạt động như một số liệu tương tự vì nó sẽ có xu hướng cao chỉ khi hai vectơ có giá trị lớn trong cùng một chiều. Ngoài ra, các vectơ có các số 0 ở các chiều khác nhau - Các vectơ trực giao - sẽ có **dot product** bằng 0, thể hiện sự khác biệt mạnh mẽ của chúng.

Tuy nhiên, **dot product** thô này có một vấn đề về một số liệu tương tự: nó ưa thích các vectơ dài. Độ dài vectơ được định nghĩa là:

$$|\vec{v}| = \sqrt{\sum_{i=1}^{N} v_i^2}$$

Dot product cao hơn nếu một vectơ dài hơn, với các giá trị cao hơn ở mỗi chiều. Các từ thường xuyên hơn có vectơ dài hơn, vì chúng có xu hướng cùng xuất hiện với nhiều từ hơn và có giá trị đồng xuất hiện cao hơn với mỗi từ đó. Do đó, **dot product** thô sẽ cao hơn đối với các từ thường xuyên. Nhưng đây là một vấn đề; chúng ta muốn một số liệu tương tự cho chúng ta biết hai từ giống nhau như thế nào bất kể tần số của chúng.

Cách đơn giản nhất để sửa đổi **dot product** để chuẩn hóa cho độ dài vectơ là chia **dot product** cho độ dài của mỗi trong hai vectơ. **Dot product** được chuẩn hóa này hóa ra giống như **cosin** của góc giữa hai vectơ, theo định nghĩa của **dot product** giữa hai vectơ a và b:

$$\begin{aligned} \vec{a} \cdot \vec{b} &= |\vec{a}||\vec{b}| \cos\theta \\ \frac{\vec{a} \cdot \vec{b}}{|\vec{a}||\vec{b}|} &= \cos\theta \end{aligned}$$

Do đó, độ tương tự cosin giữa hai vectơ v và w có thể được tính là:

$$\text{cosine}(\vec{v}, \vec{w}) = \frac{\vec{v} \cdot \vec{w}}{|\vec{v}||\vec{w}|} = \frac{\sum_{i=1}^{N} v_i w_i}{\sqrt{\sum_{i=1}^{N} v_i^2} \sqrt{\sum_{i=1}^{N} w_i^2}}$$

Đối với một số ứng dụng, chúng ta chuẩn hóa trước mỗi vectơ, bằng cách chia nó cho độ dài của nó, tạo ra một vectơ đơn vị có độ dài 1. Do đó chúng ta có thể tính toán một vectơ đơn vị từ a bằng cách chia cho $|a|$. Đối với các vectơ đơn vị, dot product giống như cosin.

Giá trị cosine nằm trong khoảng từ (0,1) đối với các vectơ chỉ cùng hướng, bằng 0 đối với các vectơ trực giao, (0, -1) đối với vectơ chỉ theo hướng ngược lại. Nhưng tần số hoặc giá trị PPMI không âm, vì vậy cosin cho các vectơ này nằm trong khoảng từ (0,1).

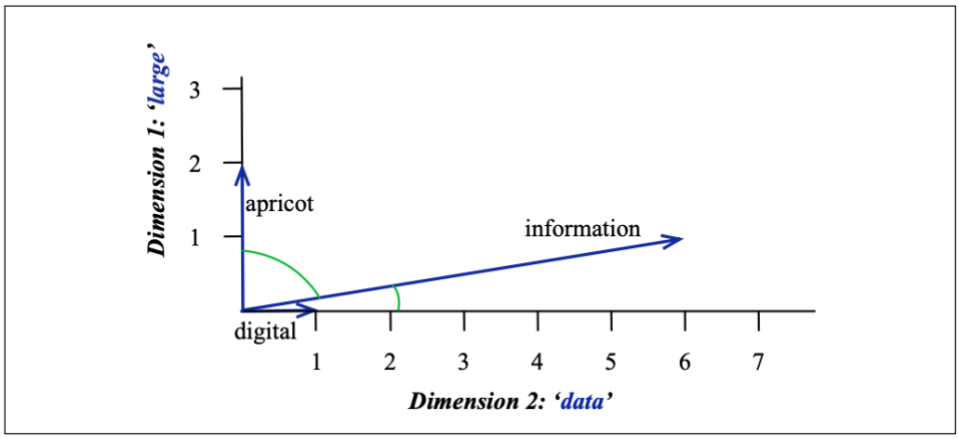
Chúng ta hãy xem cosin tính toán từ nào trong số các từ “apricot” hoặc “digital” có nghĩa gần hơn với “infomation”, chỉ sử dụng số liệu thô từ bảng đơn giản sau:

	large	data	computer
apricot	2	0	0
digital	0	1	2
information	1	6	1

$$\cos(\text{apricot}, \text{information}) = \frac{2+0+0}{\sqrt{4+0+0}\sqrt{1+36+1}} = \frac{2}{2\sqrt{38}} = .16$$

$$\cos(\text{digital}, \text{information}) = \frac{0+6+2}{\sqrt{0+1+4}\sqrt{1+36+1}} = \frac{8}{\sqrt{38}\sqrt{5}} = .58$$

Mô hình quyết định rằng "infomation" gần với "digital" hơn so với "apricot", một kết quả có vẻ hợp lý. *Hình 8* cho ta thấy một hình dung.



Hình 8:Hình minh hoạ cho phương pháp độ tương tự cosine.

Thống kê t-test, như PMI, có thể được sử dụng để đo lường mức độ thường xuyên của liên quan hơn là cơ hội. Thống kê t-test tính toán sự khác biệt giữa các phương tiện được quan sát và dự kiến, được chuẩn hóa bởi phương sai. Giá trị của t càng cao, khả năng chúng ta có thể bác bỏ giả thuyết khống cho rằng các phương tiện được quan sát và mong đợi là như nhau.

1.3.1: Các độ đo tương tự khác.

Có những lựa chọn thay thế cho độ đo cosin để đo độ tương tự. Biện pháp Jaccard (Jaccard 1908, Jaccard 1912), ban đầu được thiết kế cho các vectơ nhị phân, được Grefenstette (1994) mở rộng thành các vectơ của các sự liên kết có trọng số như sau:

$$\text{sim}_{\text{Jaccard}}(\vec{v},\vec{w}) = \frac{\sum_{i=1}^{N}\min(v_i,w_i)}{\sum_{i=1}^{N}\max(v_i,w_i)}$$

Tử số của hàm Grefenstette/Jaccard sử dụng hàm min, về cơ bản tính toán (có trọng số) số lượng các tính năng chồng chéo (vì nếu một trong hai vectơ có giá trị liên kết bằng 0 cho một thuộc tính, kết quả sẽ bằng 0). Mẫu số có thể được xem như là một yếu tố bình thường hóa (normalizing).

Độ đo Dice, được mở rộng tương tự từ vectơ nhị phân sang vectơ của các sự liên kết có trọng số; một phần mở rộng từ Curran (2003) sử dụng tử số Jaccard nhưng sử dụng hệ số chuẩn hóa mẫu số cho tổng giá trị trọng số của các mục khác không trong hai vectơ.

$$\text{sim}_{\text{Dice}}(\vec{v},\vec{w}) = \frac{2\times\sum_{i=1}^{N}\min(v_i,w_i)}{\sum_{i=1}^{N}(v_i+w_i)}$$

Cuối cùng, có một nhóm các biện pháp tương tự phân phối lý thuyết thông tin (Pereira et al. 1993, Dagan et al. 1994, Dagan et al. 1999, Lee 1999). Trực giác của các mô hình này là nếu hai vectơ, v và w, mỗi vectơ biểu thị một phân phối xác suất (giá trị của chúng bằng một), thì chúng tương tự như mức độ mà các phân phối xác suất này tương tự nhau. Cơ sở so sánh hai phân phối xác suất P và Q là **Kullback-Leibler divergence** - **KL divergence** hoặc **relative entropy** (Kullback và Leibler, 1951):

$$D(P||Q) = \sum_{x} P(x)\log\frac{P(x)}{Q(x)}$$

Thật không may, phân rã KL không được xác định khi Q (x) = 0 và P (x) = 0, đây là một vấn đề vì các vectơ phân phối từ này thường khá thưa thớt. Một cách khác (Lee, 1999) là sử dụng phân rã Jensen - Shannon, đại diện cho sự phân rã của mỗi phân phối từ giá trị trung bình của hai và không có vấn đề này với các số không.

$$JS(P||Q) = D(P|\frac{P+Q}{2}) + D(Q|\frac{P+Q}{2})$$

$$\text{sim}_{\text{JS}}(\vec{v}||\vec{w}) = D(\vec{v}|\frac{\vec{v}+\vec{w}}{2}) + D(\vec{w}|\frac{\vec{v}+\vec{w}}{2})$$

Hình 9: tóm tắt các độ đo sự liên kết và độ tương tự của vectơ mà chúng ta đã trình bày.

$$\begin{aligned}
\text{PMI}(w,f) &= \log_2 \frac{P(w,f)}{P(w)P(f)} \\
\text{t-test}(w,f) &= \frac{P(w,f)-P(w)P(f)}{\sqrt{P(f)P(w)}}
\end{aligned}$$

$$\begin{aligned}
\text{cosine}(\vec{v},\vec{w}) &= \frac{\vec{v}\cdot\vec{w}}{|\vec{v}||\vec{w}|} = \frac{\sum_{i=1}^{N} v_i \times w_i}{\sqrt{\sum_{i=1}^{N} v_i^2}\sqrt{\sum_{i=1}^{N} w_i^2}} \\
\text{Jaccard}(\vec{v},\vec{w}) &= \frac{\sum_{i=1}^{N} \min(v_i,w_i)}{\sum_{i=1}^{N} \max(v_i,w_i)} \\
\text{Dice}(\vec{v},\vec{w}) &= \frac{2\times\sum_{i=1}^{N} \min(v_i,w_i)}{\sum_{i=1}^{N} (v_i+w_i)} \\
\text{JS}(\vec{v}||\vec{w}) &= D(\vec{v}|\frac{\vec{v}+\vec{w}}{2}) + D(\vec{w}|\frac{\vec{v}+\vec{w}}{2})
\end{aligned}$$

Hình 9: Tóm tắt các độ đo.

1.4: Evaluating Vector Models.

Tất nhiên, số liệu đánh giá quan trọng nhất cho các mô hình vectơ là đánh giá bên ngoài về các nhiệm vụ; thêm chúng dưới dạng các features vào bất kỳ tác vụ NLP nào và xem liệu điều này có cải thiện hiệu suất hay không.

Tuy nhiên, nó rất hữu ích để có những đánh giá nội tại. Số liệu phổ biến nhất là kiểm tra hiệu suất của chúng về độ tương tự, và đặc biệt về tính toán mối tương quan giữa điểm tương tự từ thuật toán và xếp hạng độ tương tự từ được chỉ định bởi con người. Các bộ phán đoán khác nhau của con người giống như chúng ta mô tả trong Chương 17 về sự tương đồng dựa trên từ điển đồng nghĩa, được tóm tắt ở đây để thuận tiện. WordSim-353 (Finkelstein et al., 2002) là một bộ xếp hạng thường được sử dụng từ 0 đến 10 cho 353 cặp danh từ; ví dụ (plane, car) có điểm trung bình 5,77. SimLex-999 (Hill et al., 2015) là một bộ dữ liệu khó khăn hơn để định lượng độ tương tự (cup, mug) thay vì liên quan (cup, coffe), và bao gồm cả cặp tính từ, danh từ và động từ cụ thể và trừu tượng. Bộ dữ liệu TOEFL là một bộ gồm 80 câu hỏi, mỗi câu hỏi bao gồm một từ mục tiêu với 4 lựa chọn từ bổ sung; nhiệm vụ là chọn từ đồng nghĩa chính xác. Tất cả các bộ dữ liệu trình bày các từ không có ngữ cảnh.

Thực tế hơn một chút là các nhiệm vụ tương tự nội tại bao gồm bối cảnh. Bộ dữ liệu Tương tự từ ngữ cảnh Stanford (SCWS) (Huang và cộng sự, 2012) đưa ra một kịch bản đánh giá phong phú hơn, đưa ra phán đoán của con người về 2.003 cặp từ trong ngữ cảnh tình cảm của họ,

bao gồm danh từ, động từ và tính từ. Bộ dữ liệu này cho phép đánh giá các thuật toán tương tự từ có thể sử dụng các từ ngữ cảnh. Nhiệm vụ tương tự văn bản ngữ nghĩa (Agirre et al. 2012, Agirre et al. 2015) đánh giá hiệu suất của các thuật toán tương tự mức câu, bao gồm một tập hợp các cặp câu, mỗi cặp có điểm tương đồng được gắn nhãn người. Một nhiệm vụ khác được sử dụng để đánh giá là một nhiệm vụ tương tự, trong đó hệ thống phải giải các bài toán có dạng "a is to b as c is to d" đưa ra a, b, c và phải tìm d. Hệ thống này được đưa ra hai từ tham gia vào một mối quan hệ (ví dụ Athens và Hy Lạp, tham gia vào quan hệ thủ đô) và một từ như Oslo và phải tìm từ Na Uy. Hoặc các ví dụ định hướng cú pháp hơn: cho mouse, mice và dollar, hệ thống phải trả lại dollars. Một bộ lớn các bộ dữ liệu như vậy đã được tạo ra (Mikolov et al. 2013, Mikolov et al. 2013b)

1.5: Tổng kết.

• **Term-document matrix**, lần đầu tiên được tạo để truy xuất thông tin, có các hàng cho mỗi từ (term) trong từ vựng và một cột cho mỗi tài liệu. Ô xác định số lượng của thuật ngữ đó trong tài liệu.

• **Word-context matrix** (hoặc word-word hoặc term-term) có một hàng cho mỗi từ (mục tiêu) trong từ vựng và một cột cho mỗi từ ngữ cảnh trong từ vựng. Mỗi ô cho biết số lần thuật ngữ ngữ cảnh xảy ra trong một cửa sổ (có kích thước được chỉ định) xung quanh từ mục tiêu trong kho văn bản.

• Thay vì sử dụng ma trận đồng xuất hiện từ thô, nó thường được tính trọng số. Một trọng số phổ biến là **PPMI**.

• Trọng số thay thế là **Tf-idf**, được sử dụng cho nhiệm vụ truy xuất thông tin và các phương pháp có ý nghĩa như **t-test**.

• PPMI và các phiên bản khác của ma trận word-word có thể được xem là cung cấp các biểu diễn vectơ chiều cao, thưa thớt (hầu hết các giá trị là 0) của các từ.

• **Cosin** của hai vectơ là một hàm phổ biến được sử dụng cho sự tương tự từ (word similarity).

2. Chương 16: Semantics with Dense Vectors

Trong chương trước, chúng ta đã thấy cách biểu thị một từ như một vectơ thưa thớt với các chiều tương ứng với các từ trong từ vựng và giá trị của chúng là một số hàm đếm số từ đồng xuất hiện với mỗi từ lân cận. Do đó, mỗi từ được biểu thị bằng một vectơ dài (chiều dài | V |, với từ vựng 20.000 đến 50.000) và thưa thớt (**sparse**), với hầu hết các thành phần của vectơ cho mỗi từ bằng 0.

Trong chương này, chúng ta chuyển sang một họ các phương pháp biểu diễn từ khác: việc sử dụng các vectơ ngắn (có độ dài 50-1000) và dày đặc (**dense**) (hầu hết các giá trị đều khác không).

Các vectơ ngắn có một số lợi thế tiềm năng. Đầu tiên, chúng dễ dàng bao gồm như các feature trong các hệ thống học máy; ví dụ: nếu chúng ta sử dụng các từ nhúng 100 chiều làm feature, một bộ phân loại chỉ phải học 100 trọng số để biểu diễn một chức năng của từ, thay vì phải học hàng chục ngàn trọng số. Bởi vì chúng chứa ít tham số hơn các vectơ thưa thớt về số lượng, vectơ dense có thể khái quát tốt hơn và giúp tránh tình trạng thừa. Và các vectơ dense có thể làm tốt hơn việc nắm bắt từ đồng nghĩa so với các vectơ thưa thớt. Ví dụ, car và automobile là từ đồng nghĩa; nhưng trong một đại diện vectơ thưa thớt điển hình, các chiều của car và automobile khác biệt. Bởi vì mối quan hệ giữa chúng không được mô hình hóa, các vectơ thưa thớt có thể không nắm bắt được sự tương đồng giữa một từ như car là hàng xóm với automobile.

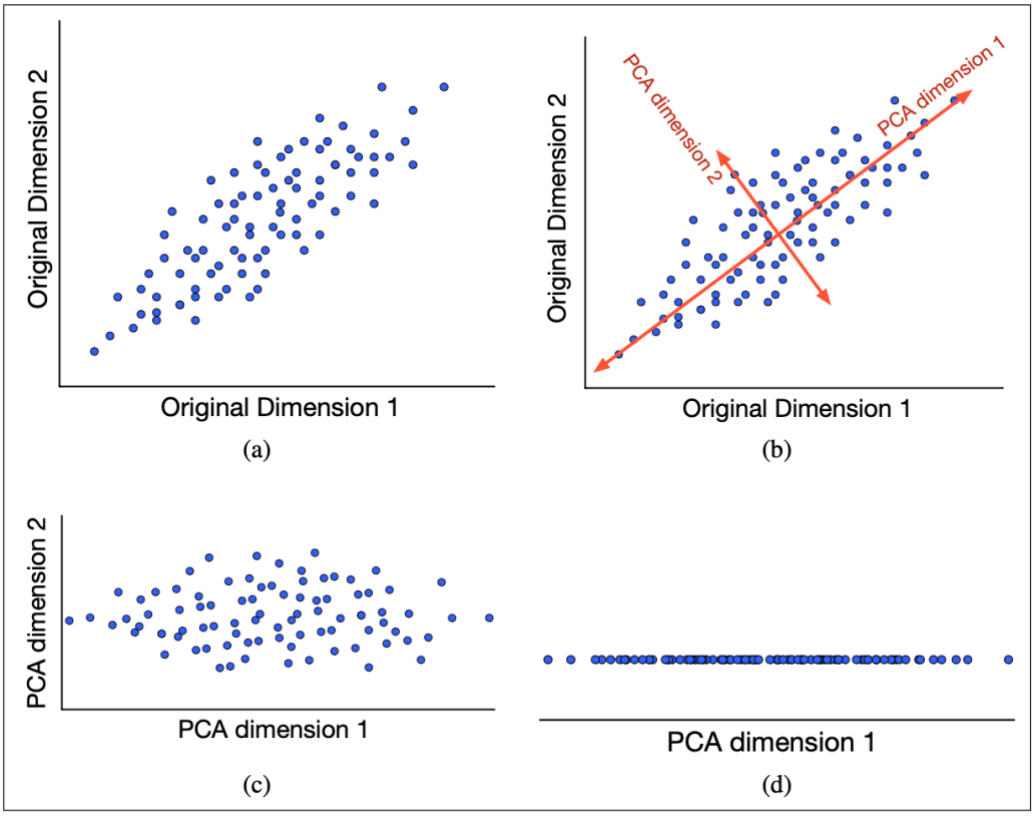
Chúng ta sẽ giới thiệu ba phương pháp tạo ra các vectơ ngắn (**dense**): (1) sử dụng các phương pháp giảm chiều như **SVD**, (2) sử dụng các mạng lưới thần kinh qua các phương pháp phổ biến như **Skip-gram** hoặc **CBOW**. (3) một cách tiếp cận khá khác dựa trên các từ lân cận được gọi là **phân cụm Brown**.

2.1: Dense Vectors via SVD

Chúng ta bắt đầu với một phương pháp cổ điển để tạo ra các **dense** vectơ : **Singular Value Decomposition**, hoặc **SVD**, lần đầu tiên được áp dụng cho nhiệm vụ tạo các embedding từ term-document matrix của Deerwester et al. (1988) trong một mô hình được gọi là Phân tích ngữ nghĩa tiềm ẩn **Latent Semantic Analysis** (**LSA**). **SVD** là một phương pháp để tìm các chiều quan trọng nhất của tập dữ liệu, mà theo các chiều đó dữ liệu thay đổi nhiều nhất. Nó có thể được áp dụng cho bất kỳ ma trận hình chữ nhật nào. **SVD** là một phần của một nhóm các phương thức có thể xấp xỉ một tập dữ liệu N chiều sử dụng ít chiều hơn, bao gồm Phân tích thành phần nguyên tắc - **Principle Components Analysis** (**PCA**), Phân tích nhân tố - **Factor Analysis**, v.v.

Nói chung, các phương pháp giảm chiều trước tiên xoay trục của tập dữ liệu gốc vào một không gian mới. Không gian mới được chọn sao cho chiều cao nhất theo sắp xếp thu được nhiều phương sai nhất trong tập dữ liệu gốc, chiều tiếp theo sẽ ghi lại phương sai cao nhất tiếp theo, v.v. *Hình 10* cho thấy một hình dung. Một tập hợp các điểm (vectơ) theo hai chiều được xoay để chiều mới đầu tiên thu được nhiều phương sai nhất trong dữ liệu. Trong không

gian mới này, chúng ta có thể biểu thị dữ liệu với số lượng kích thước nhỏ hơn (ví dụ: sử dụng một chiều thay vì hai chiều) và vẫn nắm bắt được nhiều phương sai trong dữ liệu gốc.



Hình 10: Trực quan hóa PCA.

Trực quan hóa PCA: Cho dữ liệu gốc (a) tìm thấy sự quay của dữ liệu (b) sao cho chiều thứ nhất thu được nhiều phương sai nhất và chiều thứ hai là một chiều trực giao với chiều thứ nhất. Sử dụng không gian xoay mới này (c) để thể hiện mỗi điểm trên một chiều (d). Mặc dù một số thông tin về mối quan hệ giữa các điểm ban đầu nhất thiết bị mất, chiều còn lại bảo tồn nhiều nhất mà một chiều có thể biểu diễn.

2.1.1: Latent Semantic Analysis.

LSA

Việc sử dụng **SVD** như một cách để giảm các không gian vectơ thưa thớt cho từ, giống như mô hình không gian vectơ, lần đầu tiên được áp dụng trong bối cảnh trích xuất thông tin, thường được gọi là **LSA** (phân tích ngữ nghĩa tiềm ẩn) (Deerwester et al., 1990). **LSA** là một ứng dụng cụ thể của **SVD** cho term-document matrix ($|V| \times c$) đại diện cho $|V|$ từ và sự đồng xuất hiện của chúng với các tài liệu hoặc ngữ cảnh c. **SVD** phân tích ma trận X ($|V| \times c$) thành tích của ba ma trận W, $\sum$ và C^T. Trong ma trận W ($|V| \times m$), mỗi hàng w vẫn đại diện cho một từ, nhưng các cột thì không; mỗi cột bây giờ đại diện cho một trong các kích thước m trong một không gian tiềm ẩn, sao cho các vectơ cột m trực giao với nhau và các cột được sắp xếp theo số lượng phương sai trong tập dữ liệu gốc mà. Số lượng chiều m như vậy là hạng (rank) của X (hạng của ma trận là số hàng độc lập tuyến tính). $\sum$ Là ma trận đường chéo $m \times m$, với các giá trị đơn dọc theo đường chéo, thể hiện tầm quan trọng của từng chiều. Ma trận C^T ($m \times c$) vẫn đại diện cho các tài liệu hoặc bối cảnh, nhưng mỗi hàng bây giờ đại diện cho một trong các kích thước tiềm ẩn mới và các vectơ hàng m là trực giao với nhau.

Bằng cách chỉ sử dụng k chiều đầu tiên, của W, $\sum$ và C thay vì tất cả các chiều m, tích của 3 ma trận này trở thành xấp xỉ bình phương nhỏ nhất cho X gốc. Vì các chiều đầu tiên mã hóa hầu hết phương sai, do đó việc xây dựng lại là mô hình hóa thông tin quan trọng nhất trong bộ dữ liệu gốc.

SVD áp dụng cho co-occurrence matrix X:

$$\underset{|V|\times c}{\begin{bmatrix} X \end{bmatrix}} = \underset{|V|\times m}{\begin{bmatrix} W \end{bmatrix}} \underset{m\times m}{\begin{bmatrix} \sigma_1 & 0 & 0 & \dots & 0 \\ 0 & \sigma_2 & 0 & \dots & 0 \\ 0 & 0 & \sigma_3 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & \sigma_m \end{bmatrix}} \underset{m\times c}{\begin{bmatrix} C \end{bmatrix}}$$

Chỉ lấy top k chiều (k<=m), sau khi áp dụng SVD với co-occurrence matrix X:

$$\underset{|V|\times c}{\begin{bmatrix} X \end{bmatrix}} = \underset{|V|\times k}{\begin{bmatrix} W_k \end{bmatrix}} \underset{k\times k}{\begin{bmatrix} \sigma_1 & 0 & 0 & \dots & 0 \\ 0 & \sigma_2 & 0 & \dots & 0 \\ 0 & 0 & \sigma_3 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & \sigma_k \end{bmatrix}} \underset{k\times c}{\begin{bmatrix} C \end{bmatrix}}$$

Ma trận W_k có một hàng có kích thước k cho mỗi từ có thể được sử dụng làm embedding. Hàng này bây giờ hoạt động như một vectơ k chiều đại diện cho từ đó, thay thế cho các hàng rất nhiều chiều. LSA ban đầu thường đặt k = 300, do đó, các nhúng này tương đối ngắn khi so sánh với các nhúng khác .

Thay vì trọng số PPMI hoặc tf-idf trên term-document matrix gốc, việc triển khai LSA thường sử dụng trọng số riêng của từng ô xuất hiện nhân hai trọng số được gọi là trọng số cục bộ và toàn cầu cho mỗi ô (i,j) – term i trong văn bản j.

Trọng số cục bộ của mỗi thuật ngữ i là log(tần số của nó): $\log f(i,j)+1$

Trọng số toàn cầu của thuật ngữ i là một phiên bản của entropy của nó: $1+\frac{\sum_j p(i,j)\log p(i,j)}{\log D}$ trong đó, D là số lượng văn bản.

2.1.2: SVD applied to word-context matrices

Thay vì áp dụng SVD cho term-document matrix (như trong thuật toán LSA của phần trước), một cách khác được áp dụng rộng rãi là áp dụng SVD cho word-context matrix. Trong phiên bản này, chiều ngữ cảnh là các từ chứ không phải là tài liệu, một ý tưởng đầu tiên được đề xuất bởi Schu¨tze (1992b).

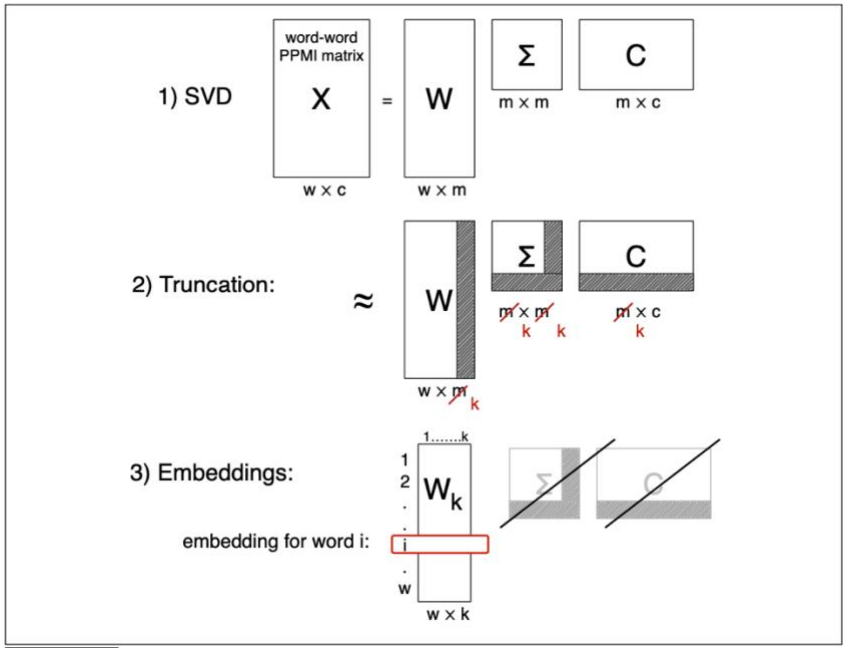
Toán học giống hệt như những gì được mô tả trong phần trước, SVD phân tích word-context matrix X thành ba ma trận W, $\sum$ và C^T. Sự khác biệt duy nhất là chúng ta đang bắt đầu

từ một word-context matrix có trọng số PPMI, thay vì term-document matrix thô. Một lần nữa, chỉ top k chiều trên cùng được giữ lại (tương ứng với các giá trị k quan trọng nhất). Cũng giống như với LSA, hàng này hoạt động như một **dense** vectơ k chiều (nhúng) đại diện cho từ đó. Các ma trận khác ($\sum$ và C) chỉ đơn giản là bị bỏ đi.

Việc sử dụng chỉ các kích thước trên cùng, cho dù đối với term-document matrix như LSA hoặc đối với word-context matrix, được gọi là **truncated SVD**. **Truncated SVD** được tham số hóa bởi k, số chiều trong biểu diễn cho mỗi từ, thường dao động từ 500 đến 5000. Do đó, SVD chạy trên word-context matrix có xu hướng sử dụng nhiều chiều hơn so với nhúng 300 chiều do LSA tạo ra. Sự khác biệt này có lẽ có liên quan đến sự khác biệt về độ chi tiết; Số lượng LSA cho các từ được phân loại nhiều hơn, đếm các lần xuất hiện trong toàn bộ tài liệu, trong khi word-context matrix PPMI đếm các từ trong một cửa sổ nhỏ. Nói chung, chiều chúng ta giữ là chiều thứ tự cao nhất, mặc dù đối với một số tác vụ, nó giúp loại bỏ một số lượng nhỏ chiều thứ tự cao nhất, chẳng hạn như 1 chiều đầu tiên hoặc thậm chí 50 chiều đầu tiên (Lapesa và Evert, 2014)

Hình 11 cho thấy một bản phác thảo cấp cao của toàn bộ quá trình SVD. Các **dense embedding** được tạo ra bởi SVD đôi khi hoạt động tốt hơn các ma trận PPMI thô trên các tác vụ ngữ nghĩa như word similarity. Các khía cạnh khác nhau của việc giảm chiều dường như đang góp phần làm tăng hiệu suất. Nếu các chiều thứ tự thấp biểu thị thông tin không quan trọng, **truncated SVD** có thể đang hoạt động để loại bỏ nhiễu. Bằng cách loại bỏ các tham số, việc cắt bớt cũng có thể giúp các mô hình tổng quát hóa tốt hơn để không nhìn thấy dữ liệu. Khi sử dụng vectơ trong các tác vụ NLP, việc có số lượng chiều nhỏ hơn có thể giúp các trình phân loại học máy dễ dàng. Và như đã đề cập ở trên, các mô hình có thể làm tốt hơn trong việc nắm bắt sự đồng xuất hiện.

Tuy nhiên, có một chi phí tính toán đáng kể cho SVD cho ma trận đồng xuất hiện lớn và hiệu suất không phải lúc nào cũng tốt hơn so với sử dụng các vectơ PPMI thưa thớt, do đó, đối với nhiều ứng dụng, vectơ thưa thớt là cách tiếp cận phù hợp. Ngoài ra, các neural mà chúng ta thảo luận trong phần tiếp theo cung cấp một giải pháp hiệu quả phổ biến để tạo ra các dense embedding.



Hình 11:Phác thảo việc sử dụng SVD từ word-word PPMI matrix.

2.2: Embeddings from prediction: Skip-gram and CBOW.

Phương pháp thứ hai để tạo ra các dense embedding lấy cảm hứng từ các mô hình mạng thần kinh được sử dụng để mô hình hóa ngôn ngữ (language modeling). Nhắc lại từ Chương 8 rằng các mô hình ngôn ngữ mạng thần kinh là đưa ra một từ và dự đoán các từ ngữ cảnh. Quá trình dự đoán này có thể được sử dụng để học các embedding cho từng từ mục tiêu. Trực giác là các từ có ý nghĩa tương tự thường xảy ra gần nhau trong các văn bản. Do đó, các mô hình thần kinh học cách nhúng bằng cách bắt đầu với một vectơ ngẫu nhiên và sau đó lặp đi lặp lại các từ nhúng của một từ để giống như các từ nhúng của các từ lân cận và ít giống như các từ nhúng không xuất hiện gần đó.

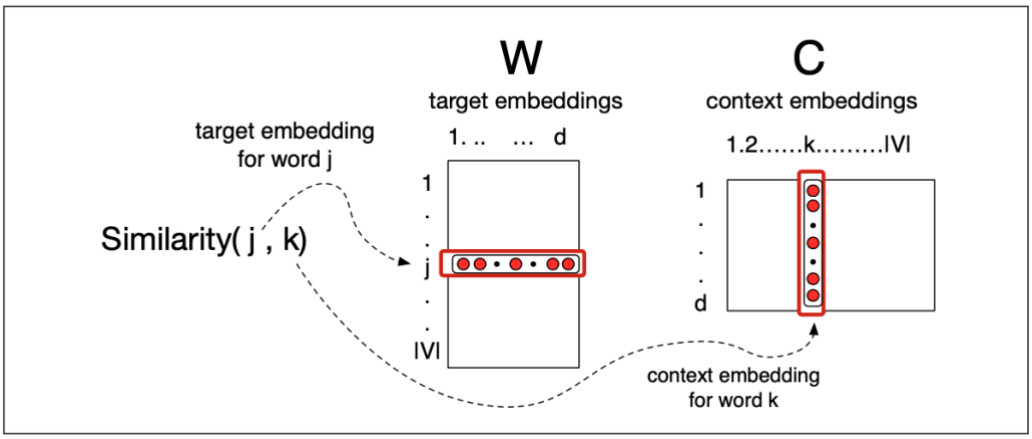
Mặc dù phép ẩn dụ cho kiến trúc này xuất phát từ dự đoán từ, chúng ta sẽ thấy rằng quá trình tìm hiểu các nhúng thần kinh này thực sự có mối quan hệ chặt chẽ với ma trận đồng xảy ra PMI, phân tích SVD.

Nhóm phương thức phổ biến nhất được gọi là word2vec, sau đó gói phần mềm thực hiện hai phương thức để tạo ra các dense embedding: Skip-gram và CBOW (Mikolov et al. 2013, Mikolov et al. 2013a). Giống như các mô hình ngôn ngữ, các mô hình word2vec học các nhúng bằng cách đào tạo một mạng để dự đoán các từ lân cận. Nhưng trong trường hợp này, nhiệm vụ dự đoán không phải là mục tiêu chính; các từ giống nhau về mặt ngữ nghĩa thường xuất hiện gần nhau trong văn bản và do đó, các từ nhúng có khả năng dự đoán các từ lân cận cũng rất tốt trong việc thể hiện sự tương đồng. Ưu điểm của các phương thức word2vec là chúng nhanh, hiệu quả để đào tạo và dễ dàng có sẵn trực tuyến với mã và các từ nhúng được xử lý trước.

Chúng ta sẽ bắt đầu với mô hình Skip-gram. Giống như mô hình SVD trong phần trước, mô hình Skip-gram thực sự học được hai cách nhúng riêng biệt cho mỗi từ w: **word embedding** v và **context embedding** c. Các nhúng này được mã hóa thành hai ma trận, ma trận từ W và ma trận ngữ cảnh C. Chúng ta sẽ thảo luận trong Phần 2.2.1 về cách học W và C, nhưng trước tiên hãy xem cách chúng được sử dụng. Mỗi hàng i của ma trận từ W là vectơ nhúng (1 × d) cho từ i trong từ vựng. Mỗi cột i của ma trận ngữ cảnh C là một vectơ (d × 1) cho từ i trong từ vựng. Về nguyên tắc, ma trận từ và ma trận ngữ cảnh có thể sử dụng các từ vựng khác nhau Vw và Vc. Tuy nhiên, trong phần còn lại của chương, chúng ta sẽ đơn giản hóa bằng cách giả sử hai ma trận chia sẻ cùng một bộ từ vựng, mà chúng ta sẽ gọi là V.

Hãy xem xét nhiệm vụ dự đoán. Chúng ta đang đi qua một kho văn bản có chiều dài T (t) và hiện đang chỉ vào từ thứ t ($w^{(t)}$), có chỉ số trong bộ từ vựng là j, vì vậy chúng ta sẽ gọi nó là w_j (1 <j <|V|). Mô hình Skip-gram dự đoán từng từ lân cận trong cửa sổ ngữ cảnh gồm 2L từ, từ hiện tại. Vì vậy, đối với cửa sổ ngữ cảnh L = 2, bối cảnh là [w^{t-2}, w^{t-1}, w^{t+1} , w^{t+2}] và chúng ta dự đoán từng từ này từ w_j. Nhưng hãy đơn giản hóa một lát và tưởng tượng chỉ dự đoán một trong những từ ngữ cảnh 2L, ví dụ w^{t+1}, có chỉ số trong từ vựng là k (1 <k <|V|). Do đó, nhiệm vụ của chúng ta là tính toán P ($w_k|w_j$).

Trung tâm của tính toán Skip-gram của xác suất p ($w_k|w_j$) là tính toán dot product giữa các vectơ cho w_k và w_j, vectơ bối cảnh cho w_k và vectơ đích cho w_j. Để đơn giản, chúng ta sẽ đại diện cho dot product này là ($c_k \cdot v_j$), (mặc dù chính xác hơn, nó phải là $c_k^T v_j$), trong đó c_k là vectơ ngữ cảnh của từ k và v_j là vectơ đích cho từ j. Như chúng ta đã thấy trong chương trước, dot product giữa hai vectơ càng cao, chúng càng giống nhau. (Đó là trực giác của việc sử dụng cosin như một thước đo tương tự, vì cosine chỉ là một dot product chuẩn hóa). *Hình 12* cho thấy trực giác rằng hàm tương tự (similarity function) yêu cầu chọn ra một vectơ đích v_j từ W và một vectơ bối cảnh c_k từ C.



Hình 12:Trực giác của similarity function.

Tất nhiên, dot product ($c_k \cdot v_j$)không phải là một xác suất, nó chỉ là một số từ đến âm đến dương vô cùng. Chúng ta có thể sử dụng hàm softmax từ Chương 7 để chuẩn hóa dot product thành xác suất. Việc tính toán mẫu số này yêu cầu tính toán dot product giữa các từ khác trong từ vựng với từ đích w_j:

$$p(w_k|w_j) = \frac{exp(c_k \cdot v_j)}{\sum_{i \in |V|} exp(c_i \cdot v_j)}$$

Tóm lại, Skip-gram tính xác suất p ($w_k|w_j$) bằng cách lấy dot product giữa vectơ từ cho j (v_j) và vectơ ngữ cảnh cho k (c_k) và biến dot product này ($v_j \cdot c_k$) thành xác suất bằng cách chuyển nó qua hàm softmax. Tuy nhiên, phiên bản thuật toán này có một vấn đề: thời gian cần thiết để tính toán mẫu số. Đối với mỗi từ w, mẫu số yêu cầu tính toán dot product với tất cả các từ khác. Như chúng ta sẽ thấy trong phần tiếp theo, chúng ta thường giải quyết điều này bằng cách sử dụng xấp xỉ mẫu số.

CBOW

Mô hình CBOW (túi từ liên tục) đại khái là hình ảnh phản chiếu của mô hình Skip-gram. Giống như Skip-gram, nó dựa trên một mô hình dự đoán, nhưng lần này dự đoán từ hiện tại w_t từ cửa sổ ngữ cảnh của các từ 2L xung quanh nó.

Mặc dù CBOW và Skip-gram là các thuật toán tương tự và tạo ra các nhúng tương tự nhau, nhưng chúng có hành vi hơi khác nhau và thường một trong số chúng sẽ trở thành lựa chọn tốt hơn cho mỗi nhiệm vụ cụ thể.

2.2.1: Learning the word and context embeddings.

Negative samples

Chúng ta đã đề cập đến trực giác để học ma trận nhúng từ W và ma trận nhúng ngữ cảnh C: lặp đi lặp lại các từ nhúng cho một từ giống nhiều hơn như các từ nhúng của hàng xóm của nó và ít hơn giống như nhúng các từ khác.

Trong phiên bản của thuật toán dự đoán được đề xuất trong phần trước, xác suất của một từ được tính bằng cách chuẩn hóa dot product giữa một từ và từng từ ngữ cảnh bởi các dot product cho tất cả các từ. Xác suất này được tối ưu hóa khi vectơ của một từ gần nhất với các từ xuất hiện gần nó (tử số) và hơn nữa từ mọi từ khác (mẫu số). Một phiên bản của thuật toán như vậy là rất tốn kém; chúng ta cần tính toán rất nhiều dot product để làm mẫu số.

Thay vào đó, phiên bản được sử dụng phổ biến nhất của Skip-gram, Skip-gram với lấy mẫu âm (negative sampling), xấp xỉ mẫu số đầy đủ này.

Phần này cung cấp một bản phác thảo ngắn gọn về cách thức này hoạt động. Trong giai đoạn huấn luyện, thuật toán đi qua kho văn bản, tại mỗi từ mục tiêu chọn các từ ngữ cảnh xung quanh làm ví dụ tích cực và đối với mỗi ví dụ tích cực cũng chọn k mẫu nhiễu (noise) hoặc mẫu âm: từ không lân cận. Mục tiêu sẽ là để di chuyển các embedding gần vào các từ hàng xóm và tránh xa các từ noise.

Ví dụ, khi đi qua văn bản ví dụ bên dưới, chúng ta đến từ "apricot" và để L = 2, chúng ta có 4 từ ngữ cảnh từ c1 đến c4:

```
lemon,  a [tablespoon of apricot preserves or] jam
          c1          c2    w     c3       c4
```

Mục tiêu là học một embedding có dot product với mỗi từ ngữ cảnh là cao. Trong thực tế Skip-gram sử dụng hàm sigmoid (σ) của dot product, trong đó $\sigma(x) = \frac{1}{1+e^{-x}}$. Vì vậy, với ví dụ trên, chúng ta muốn σ (c1 · w) + σ (c2 · w) + σ (c3 · w) + σ (c4 · w) là cao.

Ngoài ra, với mỗi từ ngữ cảnh, thuật toán chọn k từ nhiễu ngẫu nhiên theo tần số unigram của chúng. Nếu chúng ta để k = 2, cho mỗi cặp mục tiêu / bối cảnh, chúng ta sẽ có 2 từ tiếng ồn cho mỗi trong số 4 từ ngữ cảnh:

```
[cement metaphysical dear coaxial    apricot attendant whence forever puddle]
 n1      n2           n3   n4              n5        n6      n7       n8
```

Chúng ta muốn những từ noise này có dot product thấp với từ mục tiêu của chúng ta; nói cách khác, chúng ta muốn σ (n1 · w) + σ (n2 · w) + ... + σ (n8 · w) ở mức thấp.

Chính thức hơn, mục tiêu học tập cho một cặp từ / ngữ cảnh (w, c) là:

$$\log \sigma(c \cdot w) + \sum_{i=1}^{k} \mathbb{E}_{w_i \sim p(w)} \left[\log \sigma(-w_i \cdot w)\right]$$

Đó là, chúng ta muốn tối đa hóa dot product của các từ bằng ngữ cảnh thực tế và tối thiểu hóa dot product của từ đó với các từ được lấy mẫu âm k, không phải là hàng xóm. Các từ noise w_i được lấy mẫu từ bộ từ vựng V theo xác suất unigram có trọng số của chúng.

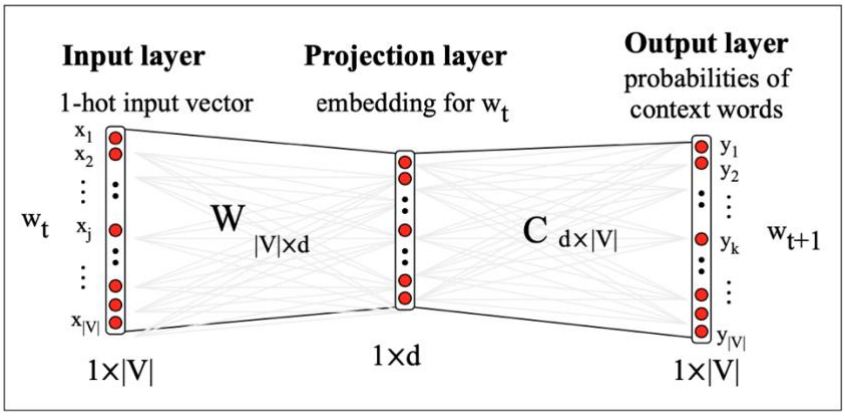
Thuật toán học bắt đầu với các ma trận W và C được khởi tạo ngẫu nhiên, sau đó đi qua tập huấn luyện di chuyển W và C để tối đa hóa phương trình trên. Một thuật toán như Stochastic Gradient Descent được sử dụng để dịch chuyển từng giá trị để tối đa hóa mục tiêu, sử dụng phương pháp truyền ngược lỗi (backpropagation) để truyền gradient trở lại qua mạng như được mô tả trong Chương 8 (Mikolov et al., 2013a).

Tóm lại, mục tiêu học tập trong phương trình trên không giống với p (w_k | w_j). Tuy nhiên, mặc dù lấy mẫu âm tính là một mục tiêu khác với mục tiêu xác suất, và do đó, các dot product kết quả sẽ không đưa ra dự đoán tối ưu cho các từ sắp tới, nó dường như tạo ra các nhúng tốt và đó là mục tiêu mà chúng ta quan tâm.

Trực quan hóa mạng

Sử dụng backpropagation lỗi yêu cầu chúng ta hình dung việc lựa chọn hai vectơ từ ma trận W và C như một mạng mà chúng ta có thể truyền ngược lại. *Hình 13* cho thấy một hình ảnh đơn giản hóa của mô hình; chúng ta đã đơn giản hóa để dự đoán một từ ngữ cảnh duy

nhất thay vì 2L các từ ngữ cảnh và đơn giản hóa để hiển thị softmax trên toàn bộ từ vựng thay vì chỉ các từ k nhiễu.



Hình 13: Đơn giản hoá mô hình Skip-gram.

Thật đáng để dành một chút thời gian để hình dung cách mạng đang tính toán xác suất giống như phiên bản dot product mà chúng ta đã mô tả ở trên. Trong mạng của *Hình 13*, chúng ta bắt đầu với một vectơ đầu vào x, là một one-hot vectơ cho từ hiện tại w_j. One-hot Vectơ là một vectơ có một phần tử bằng 1 và tất cả các phần tử khác được đặt thành 0.

Sau đó, chúng ta dự đoán xác suất của mỗi từ đầu ra trong 2L từ. Có nghĩa là một từ đầu ra w_{t+1}, trong 3 bước:

1. **Chọn embedding từ W**: x được nhân với W, ma trận đầu vào, để đưa ra lớp ẩn hoặc Projection Layer. Vì mỗi hàng của ma trận đầu vào W chỉ là một embedding cho từ w_t và đầu vào là one-hot cho w_j, nên Projection Layer cho đầu vào x sẽ là h = W * w_j = v_j, input embedding cho w_j.

2. **Tính toán dot product $c_k \cdot v_j$**: Với mỗi từ ngữ cảnh trong 2L, bây giờ chúng ta nhân vectơ chiếu h với ma trận ngữ cảnh C. Kết quả cho mỗi từ ngữ cảnh, o = Ch, là vector đầu ra (1 × |V|) chiều cho mỗi từ trong |V| từ vựng. Khi làm như vậy, phần tử o_k được tính bằng cách nhân h với nhúng đầu ra cho từ w_k: $o_k = c_k \cdot h = ck \cdot v_j$.

3. **Normalize dot product thành xác suất**: Đối với mỗi từ ngữ cảnh, chúng ta bình thường hóa vectơ dot product này, biến từng điểm thành phần o_k thành xác suất bằng cách sử dụng hàm soft-max:

$$p(w_k|w_j) = y_k = \frac{exp(c_k \cdot v_j)}{\sum_{i \in |V|} exp(c_i \cdot v_j)}$$

2.2.2: Mối quan hệ giữa các loại embedding khác nhau.

Có một mối quan hệ thú vị giữa Skip-gram, SVD / LSA và PPMI. Nếu chúng ta nhân hai ma trận bối cảnh WC, chúng ta tạo ra một ma trận X (|V| × |V|), mỗi mục x_{ij} tương ứng với một số liên kết giữa từ đầu vào i và từ ngữ cảnh j. Levy và Goldberg (2014b) chứng minh rằng giá trị tối ưu của Skip-gram xảy ra khi ma trận đã học này thực sự là một phiên bản của ma trận PMI, với các giá trị được thay đổi bởi log k (trong đó k là số lượng mẫu âm trong Skip-gram with negative sampling):

$$WC = X^{\text{PMI}} - \log k$$

Nói cách khác, Skip-gram đang ngầm định phân tích một ma trận PMI (phiên bản đã thay đổi) thành hai ma trận nhúng W và C, giống như SVD đã làm. Xem Levy và Goldberg (2014b) để biết thêm chi tiết.

Khi các phần nhúng được học, chúng ta sẽ có hai phần nhúng cho mỗi từ w_i: v_i và c_i. Chúng ta có thể chọn loại bỏ ma trận C và chỉ giữ W, như chúng ta đã làm với SVD, trong trường hợp đó, mỗi từ i sẽ được biểu thị bằng vectơ v_i.

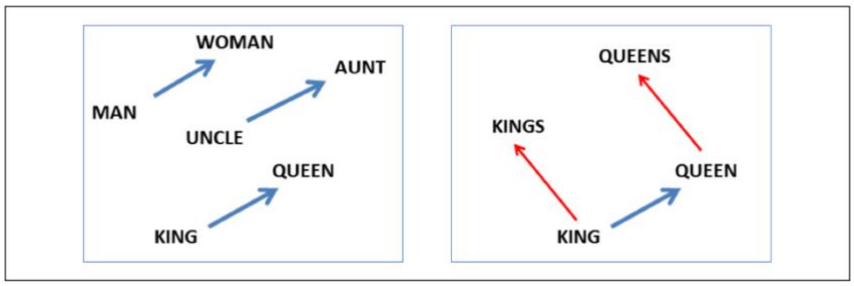
Ngoài ra, chúng ta có thể cộng hai phần nhúng với nhau, được vector nhúng d chiều mới, hoặc chúng ta có thể ghép chúng thành một để được vector nhúng 2d chiều.

Như với các phương thức dựa trên số đếm đơn giản như PPMI, kích thước cửa sổ ngữ cảnh L ảnh hưởng đến hiệu suất của skip-gram và các thử nghiệm thường điều chỉnh tham số L trên tập hợp dev. Cũng như PPMI, kích thước cửa sổ dẫn đến sự khác biệt về chất lượng: cửa sổ nhỏ hơn nắm bắt nhiều thông tin cú pháp hơn, cửa sổ lớn hơn thông tin ngữ nghĩa và quan hệ. Một điểm khác biệt so với các phương pháp dựa trên số đếm là đối với Skip-gram, kích thước cửa sổ càng lớn thì thuật toán yêu cầu đào tạo càng nhiều (phải dự đoán nhiều từ lân cận hơn). Xem phần cuối của chương để biết con trỏ đến các khảo sát đã khám phá các tham số hóa như windowsize cho các tác vụ khác nhau.

2.3: Thuộc tính của embeddings.

Một thuộc tính ngữ nghĩa của các loại nhúng khác nhau có thể dùng trong tính hữu dụng của chúng là khả năng nắm bắt các ý nghĩa quan hệ

Mikolov và cộng sự. (2013b) chứng minh rằng độ lệch giữa các vectơ nhúng có thể nắm bắt một số mối quan hệ giữa các từ, ví dụ: kết quả của vectơ xuất phát ('king') - vectơ ('man') + vectơ ('woman') là một vectơ gần với vectơ ('queen); Tương tự, họ thấy rằng vectơ biểu thức ('Paris') - vectơ ('Pháp') + vectơ ('Ý') dẫn đến một vectơ rất gần với vecni ('Rome'). Levy và Goldberg (2014a) cho thấy nhiều loại hình nhúng khác dường như cũng có tính chất này.



Hình 14: Tính chất của embeddings.

2.4: Brown Clustering.

Phân cụm Brown (Brown et al., 1992) là một thuật toán phân cụm kết tụ để lấy các biểu diễn vectơ của các từ bằng cách phân cụm các từ dựa trên sự liên kết của chúng với các từ trước hoặc sau.

Thuật toán sử dụng mô hình ngôn ngữ dựa trên lớp (**class-based language model**) (Brown et al., 1992), một mô hình trong đó mỗi từ $w \in V$ thuộc về một lớp $c \in C$ với xác suất $P(w|c)$. **Class-based language model** gán xác suất cho một cặp từ wi-1 và wi bằng cách mô hình hóa quá trình chuyển đổi giữa các lớp thay vì giữa các từ:

$$P(w_i|w_{i-1}) = P(c_i|c_{i-1})P(w_i|c_i)$$

Class-based language model có thể được sử dụng để gán xác suất cho toàn bộ kho dữ liệu được cung cấp một cụm C đặc biệt như sau:

$$P(\text{corpus}|C) = \prod_{i-1}^{n} P(c_i|c_{i-1})P(w_i|c_i)$$

Class-based language model thường không được sử dụng làm mô hình ngôn ngữ cho các ứng dụng như dịch máy hoặc nhận dạng giọng nói vì chúng không hoạt động tốt như các mô hình ngôn ngữ n-gram hoặc thần kinh tiêu chuẩn. Nhưng chúng là một thành phần quan trọng trong phân cụm Brown.

Phân cụm Brown là một thuật toán phân cụm phân cấp (hierarchical clustering algorithm). Chúng ta hãy xem xét một phiên bản ngây thơ (mặc dù không hiệu quả) của thuật toán:

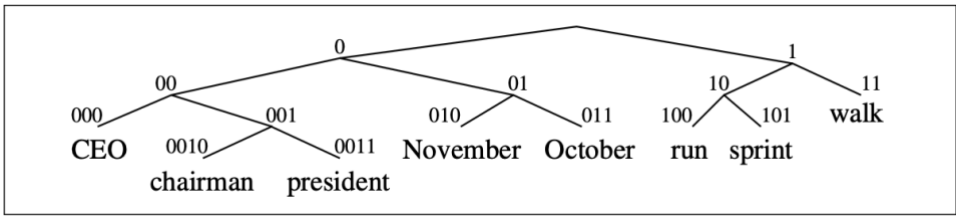
1. Mỗi từ ban đầu được gán cho cụm riêng của nó.

2. Bây giờ chúng ta xem xét hợp nhất từng cặp cụm. Cặp có sự hợp nhất dẫn đến sự giảm nhỏ nhất về khả năng của kho văn bản (theo mô hình ngôn ngữ dựa trên lớp) được hợp nhất.

3. Phân cụm tiến hành cho đến khi tất cả các từ nằm trong một cụm lớn.

Do đó, hai từ rất có thể được phân cụm nếu chúng có xác suất tương tự cho các từ trước và sau, dẫn đến các cụm mạch lạc hơn. Kết quả là các từ sẽ được hợp nhất nếu chúng giống nhau theo ngữ cảnh.

Bằng cách truy tìm thứ tự các cụm được hợp nhất, mô hình xây dựng một cây nhị phân từ dưới lên trên, trong đó các lá là các từ trong từ vựng và mỗi nút trung gian trong cây đại diện cho cụm được hình thành bằng cách hợp nhất các con của nó. *Hình 15* cho thấy một sơ đồ của một phần của cây.



Hình 15: Brown clustering là một binary tree.

Sau khi phân cụm, một từ có thể được biểu diễn bằng chuỗi nhị phân tương ứng với đường dẫn của nó từ nút gốc; 0 cho trái, 1 cho phải, tại mỗi điểm lựa chọn trong cây nhị phân. Ví dụ, trong hình 16.9, từ chủ tịch là vectơ 0010 và tháng 10 là 011. Vì cụm màu nâu là thuật toán phân cụm cứng (mỗi từ chỉ có một cụm), chỉ có một chuỗi trên mỗi từ. Bây giờ chúng ta có thể trích xuất các tính năng hữu ích bằng cách lấy các tiền tố nhị phân của chuỗi bit này; mỗi tiền tố đại diện cho một cụm mà từ đó thuộc về. Ví dụ: chuỗi 01 trong hình biểu thị cụm tên tháng {tháng 11, tháng 10}, chuỗi 0001 tên của danh từ chung cho giám đốc điều hành công ty {chủ tịch, chủ tịch}, 1 là động từ {run, sprint, walk} và 0 là danh từ. Các tiền tố này sau đó có thể được sử dụng như một đại diện vector cho từ; tiền tố càng ngắn, cụm càng trừu tượng. Do đó, độ dài của biểu diễn vector có thể được điều chỉnh để phù hợp với nhu cầu của nhiệm vụ cụ thể. Koo và cộng sự. (2008) cải thiện phân tích cú pháp bằng cách sử dụng nhiều tính năng: tiền tố 4 - 6 bit để nắm bắt một phần thông tin giọng nói và chuỗi bit đầy đủ để thể hiện các từ. Spitkovsky và cộng sự. (2011) cho thấy các vectơ được tạo từ 8 hoặc 9 bit đầu tiên của cụm Brown hoạt động tốt khi cảm ứng ngữ pháp. Bởi vì chúng dựa trên các từ lân cận ngay lập tức, cụm Brown được sử dụng phổ biến nhất để thể hiện các thuộc tính cú pháp của từ và do đó thường được sử dụng như một tính năng trong trình phân tích cú pháp. Tuy nhiên, các cụm cũng đại diện cho một số thuộc tính ngữ nghĩa. Hình 16.10 cho thấy một số ví dụ từ một cụm lớn từ Brown et al. (1992).

Lưu ý rằng phiên bản ngây thơ của thuật toán phân cụm Brown được mô tả ở trên cực kỳ kém hiệu quả - O (n5): tại mỗi lần lặp n, thuật toán xem xét từng phép hợp nhất của n2 và với mỗi phép hợp nhất, tính giá trị của phân cụm bằng cách tính tổng qua n2 điều kiện. bởi vì nó

phải xem xét mọi cặp hợp nhất có thể. Trong thực tế, chúng ta sử dụng các thuật toán O (n3) hiệu quả hơn, sử dụng các bảng để tính toán trước các giá trị cho mỗi hợp nhất (Brown et al. 1992, Liang 2005).

2.5: Summary.

• **Singular Value Decomposition** (**SVD**) là một kỹ thuật có thể được sử dụng để tạo các embedding có số chiều thấp hơn từ term-document matrix hoặc term-term matrix.

• **Phân tích ngữ nghĩa tiềm ẩn** (**LSA**) là một ứng dụng của SVD cho term-document matrix, sử dụng các trọng số cụ thể và dẫn đến các nhúng có khoảng 300 chiều.

• Hai thuật toán lấy cảm hứng từ các mô hình ngôn ngữ, **Skip-gram** và **CBOW**, là những cách hiệu quả phổ biến để tính toán các embedding. Chúng học các embedding (theo cách ban đầu được lấy cảm hứng từ tài liệu dự đoán từ mạng thần kinh) bằng cách tìm các nhúng có một dot product cao với các từ lân cận và dot product thấp với các từ noise.

• **Phân cụm Brown** là một phương pháp nhóm các từ thành cụm dựa trên mối quan hệ của chúng với các từ trước và sau. **Phân cụm Brown** có thể được sử dụng để tạo các vectơ bit cho một từ có thể hoạt động như một biểu diễn cú pháp.

3. Chương 17: Computing with Word Senses

Hai chương trước tập trung vào các cách trình bày ý nghĩa cho toàn bộ câu. Trong các cuộc thảo luận đó, chúng ta đã đưa ra một giả định đơn giản hóa bằng cách biểu diễn nghĩa của từ giống như các ký hiệu không được phân tích như EAT hoặc JOHN hoặc RED. Nhưng biểu diễn cho ý nghĩa của một từ bằng cách viết hoa nó là một mô hình khá không đạt yêu cầu. Trong chương này, chúng ta giới thiệu một mô hình phong phú hơn về ngữ nghĩa của các từ, dựa trên nghiên cứu ngôn ngữ về nghĩa của từ, một lĩnh vực được gọi là ngữ nghĩa từ vựng (**lexical semantics**), cũng như nghiên cứu tính toán của các nghĩa này.

Để biểu thị nghĩa của từ, chúng ta sẽ bắt đầu bằng hình thức từ gốc (**lemma**) mà chúng ta đã nói trong Chương 4, là hình thức ngữ pháp của một từ được sử dụng để thể hiện một từ trong từ điển và từ điển đồng nghĩa. Do đó, carpet là **lemma** cho carpets, và sing là **lemma** cho sing, sang, sung. Trong nhiều ngôn ngữ, hình thức nguyên bản được sử dụng làm lemma cho động từ, do đó, “dormir” Tây Ban Nha (to sleep) là **lemma** cho “duermes” (you sleep. Các hình thức cụ thể “sung” hoặc “carpets” hoặc “sing” hoặc “duermes” được gọi là **wordforms**.

Nhưng một **lemma** vẫn có thể có nhiều ý nghĩa khác nhau. Lemma “bank” có thể đề cập đến một tổ chức tài chính hoặc đến sườn dốc của một con sông. Chúng ta gọi mỗi khía cạnh trong ý nghĩa của “bank” là một **word sense**. Thực tế là **lemma** có thể là từ đồng âm (**homonymous**) (có nhiều sense) gây ra tất cả các loại vấn đề trong xử lý văn bản. Định nghĩa từ ngữ (**Word sense disambiguation**) là nhiệm vụ xác định nghĩa của từ đang được sử dụng trong một ngữ cảnh cụ thể, một nhiệm vụ có lịch sử lâu dài trong ngôn ngữ học tính toán và các nhiệm vụ ứng dụng từ dịch máy đến trả lời câu hỏi. Chúng ta đưa ra một số thuật toán để sử dụng các feature từ ngữ cảnh để quyết định ý nghĩa nào được dự định trong một bối cảnh cụ thể.

Chúng ta cũng sẽ giới thiệu **WordNet**, một từ điển đồng nghĩa được sử dụng rộng rãi để đại diện cho các **word sense** và đại diện cho các mối quan hệ giữa các sense, như mối quan hệ IS-A giữa “chó” và “động vật có vú” hoặc mối quan part-whole giữa “xe hơi” và “động cơ”. Cuối cùng, chúng ta sẽ giới thiệu nhiệm vụ tính toán độ tương tự của từ và chỉ ra cách sử dụng từ điển đồng nghĩa dựa trên cảm giác (sense) như WordNet để quyết định xem hai từ có nghĩa tương tự hay không.

3.1: Word Senses.

Hãy xem xét hai cách sử dụng lemma “bank” được đề cập ở trên, có nghĩa là một cái gì đó như 'tổ chức tài chính' và 'gò dốc', tương ứng:

(1): Instead, a bank can hold the investments in a custodial account in the client’s name.

(2): But as agriculture burgeons on the east bank, the river will shrink even more.

Chúng ta đại diện cho sự thay đổi trong cách sử dụng bằng cách nói rằng lemma “bank” có hai sense. Một sense (hoặc word sense) là một đại diện rời rạc của một khía cạnh của ý nghĩa của một từ. Theo truyền thống từ vựng một cách lỏng lẻo, chúng ta biểu diễn cho mỗi sense bằng cách đặt một siêu ký tự, ví dụ như trong bank[1] và bank.

Các sense của một từ có thể không có bất kỳ mối quan hệ cụ thể nào giữa chúng; có thể gần như ngẫu nhiên khi họ chia sẻ một hình thức chỉnh hình. Ví dụ, tổ chức tài chính và dốc của “bank” dường như không liên quan. Trong những trường hợp như vậy, chúng ta nói rằng hai sense là từ đồng âm (**homonyms**) và mối quan hệ giữa các sense là một từ đồng âm (**homonymy**). Do đó, bank[1] ('tổ chức tài chính') và bank[2] ('gò dốc') là những từ đồng âm. Chúng ta nói rằng hai cách sử dụng “bank” này là đồng âm, bởi vì chúng được viết giống nhau.

Hai từ có thể là từ đồng âm theo một cách khác nếu chúng được đánh vần khác nhau nhưng phát âm giống nhau, ví dụ: “write” và “right”, hoặc “piece” và “peace”. Chúng ta gọi những từ đồng âm này là **homophones** và chúng ta đã thấy trong Ch. 5 rằng **homophones** là một trong những nguyên nhân gây ra lỗi chính tả.

Đôi khi cũng có một số kết nối ngữ nghĩa giữa các sense của một từ. Hãy xem xét ví dụ sau:

(3) While some banks furnish blood only to hospitals, others are less restrictive.

Mặc dù đây rõ ràng không phải là cách sử dụng ý nghĩa 'gò dốc' của “bank”, nhưng rõ ràng nó không phải là một tài liệu tham khảo cho một hoạt động từ thiện của một tổ chức tài chính. Thay vào đó, bank có toàn bộ phạm vi sử dụng liên quan đến kho lưu trữ cho các thực thể sinh học khác nhau, như trong ngân hàng máu, ngân hàng trứng và ngân hàng tinh trùng. Vì vậy, chúng ta có thể gọi đây là "ngân hàng sinh học" nghĩa là bank^3. Bây giờ ý nghĩa mới này bank^3 có một số loại quan hệ với bank^1; cả bank^1 và bank^3 đều là kho lưu trữ cho các thực thể có thể được gửi và lấy ra; trong bank^1 thực thể là tiền tệ, trong khi ở bank^3 thực thể là sinh học.

Khi hai sense có liên quan về mặt ngữ nghĩa, chúng ta gọi mối quan hệ giữa chúng là đa nghĩa (**polysemy**) chứ không phải là đồng âm. Trong nhiều trường hợp của **polysemy**, mối quan hệ ngữ nghĩa giữa các sense là có hệ thống và có cấu trúc. Ví dụ, hãy xem xét một ý nghĩa khác của “bank”, được minh họa trong câu sau:

(4) The bank is on the corner of Nassau and Witherspoon.

Ý nghĩa này, mà chúng ta có thể gọi là bank^4, có nghĩa là "tòa nhà thuộc về một tổ chức tài chính". Nó chỉ ra rằng hai loại sense này (một tổ chức và tòa nhà liên kết với một tổ chức) xảy ra cùng với nhiều từ khác (trường học, trường đại học, bệnh viện, v.v.). Do đó, có một mối quan hệ có hệ thống giữa các sense mà chúng ta có thể đại diện là:

BUILDING ↔ ORGANIZATION

Kiểu con đặc biệt này của mối quan hệ đa nghĩa thường được gọi là hoán dụ. Hoán dụ là việc sử dụng một khía cạnh của một khái niệm hoặc thực thể để chỉ các khía cạnh khác của thực thể hoặc cho chính thực thể đó. Do đó, chúng tôi đang thực hiện hoán dụ khi chúng tôi sử dụng cụm từ Nhà Trắng để chỉ chính quyền có văn phòng ở Nhà Trắng. Các ví dụ phổ biến khác về hoán dụ bao gồm mối quan hệ giữa các cặp sense sau:

Author (*Jane Austen wrote Emma*) ↔ Works of Author (*I really love Jane Austen*)
Tree (*Plums have beautiful blossoms*) ↔ Fruit (*I ate a preserved plum yesterday*)

Mặc dù có thể hữu ích để phân biệt **polysemy** với đồng âm không liên quan, nhưng không có ngưỡng cứng nào cho việc hai sense liên quan phải được coi là đa nghĩa. Vì vậy, sự khác biệt thực sự là một mức độ. Thực tế này có thể làm cho rất khó để quyết định một từ có bao nhiêu sense, nghĩa là có tạo ra các sense riêng biệt cho các cách sử dụng liên quan chặt chẽ hay không. Có nhiều tiêu chí khác nhau để quyết định rằng các cách sử dụng khác nhau của một từ nên được biểu diễn dưới dạng các sense riêng biệt. Chúng ta có thể xem xét hai sense rời rạc nếu chúng có điều kiện chân lý độc lập, hành vi cú pháp khác nhau và quan hệ ý thức độc lập hoặc nếu chúng thể hiện ý nghĩa đối kháng.

Hãy xem xét các cách sử dụng sau của động từ “*serve*” từ kho dữ liệu WSJ:

(5) They rarely *serve* red meat, preferring to prepare seafood.

(6) He *served* as U.S. ambassador to Norway in 1976 and 1977.

(7) He might have *served* his time, come out and led an upstanding life.

"serve" of serving red meat và serving time rõ ràng có các điều kiện và giả định khác nhau; serve of serve as ambassador có cấu trúc phân loại riêng biệt như serve as NP. Các heuristic cho thấy đây có lẽ là ba sense riêng biệt của serve. Một kỹ thuật thực tế để xác định xem hai sense có khác biệt hay không là kết hợp hai cách sử dụng một từ trong một câu; loại kết hợp của các bài đọc đối kháng được gọi là **zeugma**. Hãy xem xét các ví dụ ATIS sau:

(8) Which of those flights serve breakfast?
(9) Does Midwest Express serve Philadelphia?
(10) ?Does Midwest Express serve breakfast and Philadelphia?

Chúng ta sử dụng (?) Để đánh dấu những ví dụ được hình thành về mặt ngữ nghĩa. Sự kỳ lạ của ví dụ thứ ba được tìm thấy (một trường hợp của **zeugma**) cho thấy không có cách hợp lý nào để tạo cảm giác phục vụ công việc duy nhất cho cả bữa sáng và Philadelphia. Chúng ta có thể sử dụng điều này như bằng chứng cho thấy phục vụ có hai giác quan khác nhau trong trường hợp này.

Từ điển có xu hướng sử dụng nhiều giác quan tinh tế để nắm bắt sự khác biệt về ý nghĩa tinh tế, một cách tiếp cận hợp lý cho rằng vai trò truyền thống của từ điển là hỗ trợ người học từ. Đối với mục đích tính toán, chúng ta thường không cần những sự phân biệt tốt đẹp này, vì vậy chúng ta có thể muốn nhóm hoặc nhóm các sense; chúng ta đã thực hiện điều này cho một số ví dụ trong chương này.

Làm thế nào chúng ta có thể định nghĩa word sense? Chúng ta có thể tìm trong từ điển không? Hãy xem xét các đoạn sau từ các định nghĩa của right, left, red và blood từ American Heritage Dictionary (Morris, 1985).

right	*adj.* located nearer the right hand esp. being on the right when facing the same direction as the observer.
left	*adj.* located nearer to this side of the body than the right.
red	*n.* the color of blood or a ruby.
blood	*n.* the red liquid that circulates in the heart, arteries and veins of animals.

Lưu ý tính tuần hoàn trong các định nghĩa này. Định nghĩa "right" tạo ra hai tham chiếu trực tiếp đến chính nó và "left" chứa một tham chiếu tự ẩn trong cụm từ "this side of the body". Các mục cho "red" và "blood" tránh loại tự tham chiếu trực tiếp này bằng cách thay vào đó tham chiếu lẫn nhau trong định nghĩa của chúng. Tính tuần hoàn như vậy dĩ nhiên là cố hữu trong tất cả các định nghĩa từ điển; những ví dụ này chỉ là những trường hợp cực đoan. Đối với con người, những mục như vậy vẫn hữu ích vì người dùng từ điển đã nắm bắt đủ các thuật ngữ khác.

Đối với mục đích tính toán, một cách tiếp cận để xác định sense là sử dụng một cách tiếp cận tương tự với các định nghĩa từ điển này; xác định một sense thông qua mối quan hệ của nó với các sense khác. Ví dụ, các định nghĩa ở trên cho thấy rõ rằng "right" và "left" là các loại lemma tương tự đứng trong một loại thay thế hoặc đối lập với nhau. Tương tự như vậy, chúng ta có thể lượm lặt rằng "red" là một màu, nó có thể được áp dụng cho cả "blood" và "rubies", và "blood" là một chất lỏng. **Sense relations** của loại này được thể hiện trong cơ sở dữ liệu trực tuyến như **WordNet**. Với một cơ sở dữ liệu đủ lớn về các mối quan hệ như vậy, nhiều ứng dụng hoàn toàn có khả năng thực hiện các tác vụ ngữ nghĩa tinh vi (ngay cả khi chúng không thực sự biết bên phải của nó từ bên trái).

3.2: Relations Between Senses.

Phần này khám phá một số mối quan hệ giữa các word sense, tập trung vào một số mối quan hệ quan trọng như: từ đồng nghĩa (**synonymy**), trái nghĩa (**antonymy**) và từ có nghĩa khái quát chung (**hypernymy**), cũng như đề cập ngắn gọn về các mối quan hệ khác như từ bộ phận để chỉ toàn bộ (**meronymy**).

3.2.1: Đồng nghĩa (Synonymy) và trái nghĩa (Antonymy).

Synonymy

Khi hai word sense của hai từ (lemma) khác nhau giống hệt nhau hoặc gần giống nhau, chúng ta nói hai word sense là từ đồng nghĩa (**Synonymy**). **Synonymy** bao gồm các cặp như

couch/sofa vomit/throw up filbert/hazelnut car/automobile

Một định nghĩa chính thức hơn về **Synonymy** (giữa các từ chứ không phải là word sense) là hai từ đồng nghĩa nếu chúng thay thế cho từ kia trong bất kỳ câu nào mà không thay đổi các điều kiện thật của câu. Chúng ta thường nói trong trường hợp này rằng hai từ có cùng một nghĩa mệnh đề (**propositional meaning**).

Trong khi sự thay thế giữa một số cặp từ như car/automobile hoặc water/H2O là sự bảo tồn sự thật, thì các từ này vẫn không giống nhau về nghĩa. Thật vậy, có lẽ không có hai từ nào hoàn toàn giống nhau về nghĩa và nếu chúng ta định nghĩa từ đồng nghĩa là ý nghĩa và ý nghĩa giống hệt nhau trong tất cả các ngữ cảnh, có lẽ không có từ đồng nghĩa tuyệt đối. Bên cạnh ý nghĩa mệnh đề (**propositional meaning**), nhiều khía cạnh khác của ý nghĩa phân biệt các từ này rất quan trọng. Ví dụ, H2O được sử dụng trong bối cảnh khoa học và sẽ không phù hợp trong hướng dẫn đi bộ đường dài; sự khác biệt trong thể loại này là một phần của ý nghĩa của từ. Trong thực tế, từ đồng nghĩa từ thường được sử dụng để mô tả mối quan hệ của từ đồng nghĩa gần đúng hoặc thô.

Synonymy thực sự là một mối quan hệ giữa các word sense hơn là word. Xem xét các từ "big" và "large". Đây có vẻ là từ đồng nghĩa trong các câu ATIS sau đây, vì chúng ta có thể hoán đổi big và large trong cả hai câu và giữ nguyên nghĩa:

How big is that plane?
Would I be flying on a large or small plane?

Nhưng lưu ý câu WSJ sau đây trong đó chúng ta không thể thay thế "big" với "large":

Miss Nelson, for instance, became a kind of big sister to Benjamin.
?Miss Nelson, for instance, became a kind of large sister to Benjamin.

Điều này là do từ "big" có word sense là già hoặc lớn lên, trong khi "large" lại thiếu nghĩa này. Vì vậy, chúng ta nói rằng một số word sense của "big" và "large" là đồng nghĩa (gần nhau) trong một số trường hợp khác thì không.

Antonym

Từ đồng nghĩa là những từ có nghĩa giống hoặc tương tự. Từ trái nghĩa (**antonym**), ngược lại, là những từ có nghĩa trái ngược nhau, ví dụ như sau:

long/short big/little fast/slow cold/hot dark/light
rise/fall up/down in/out

Hai word sense có thể là từ trái nghĩa nếu chúng xác định đối lập nhị phân hoặc ở hai đầu đối diện của một số thang đo. Đây là trường của hợp long/short, fast/slow hoặc big/little;. Một nhóm từ trái nghĩa khác, đảo ngược (**reversives**), mô tả sự thay đổi hoặc chuyển động theo hướng ngược lại, chẳng hạn như rise/fall hoặc up/down.

Do đó, các từ trái nghĩa hoàn toàn khác nhau về một khía cạnh trong ý nghĩa của chúng. Vị trí của chúng trên thang đo hoặc hướng đi của chúng nhưng lại rất giống nhau, chia sẻ gần như tất cả các khía cạnh khác về ý nghĩa. Do đó, việc tự động phân biệt từ đồng nghĩa (**Synonymy**) với từ trái nghĩa (**Antonym**) có thể khó khăn.

3.2.2: Hyponymy

Một word sense là một **hyponym** của word sense khác nếu ý nghĩa đầu tiên cụ thể hơn, biểu thị một lớp con của word sense khác. Ví dụ, "car" là một **hyponym** của "vehicle"; "dog" là một **hyponym** của "animal", và "mango" là một **hyponym** của "fruit".

Ngược lại, chúng ta nói rằng "vehicle" là một **hypernym** của "car", và "animal" là một **hypernym** của "dog". Thật không may là hai từ (hypernym và hyponym) rất giống nhau và do đó dễ bị nhầm lẫn; vì lý do này, từ **superordinate** thường được sử dụng thay vì **hypernym**.

couch/sofa vomit/throw up filbert/hazelnut car/automobile

Một định nghĩa chính thức hơn về **Synonymy** (giữa các từ chứ không phải là word sense) là hai từ đồng nghĩa nếu chúng thay thế cho từ kia trong bất kỳ câu nào mà không thay đổi các điều kiện thật của câu. Chúng ta thường nói trong trường hợp này rằng hai từ có cùng một nghĩa mệnh đề (**propositional meaning**).

Superordinate	vehicle	fruit	furniture	mammal
Hyponym	car	mango	chair	dog

Hyponymy thường là một mối quan hệ bắc cầu; nếu A là một **hyponym** của B và B là một **hyponym** của C, thì A là một **hyponym** của C.

Meronymy

Một mối quan hệ phổ biến khác là **meronymy**, quan hệ một phần. Một "leg" là một phần của "chair"; một "wheel" là một phần của "car". Chúng tôi nói rằng "wheel" là một **meronymy** của car, và car là một **holonym** của "wheel".

3.3: WordNet: Cơ sở dữ liệu về quan hệ từ điển (Lexical Relations).

Tài nguyên được sử dụng phổ biến nhất cho quan hệ nghĩa tiếng Anh là cơ sở dữ liệu từ vựng **WordNet** (Fellbaum, 1998). **WordNet** bao gồm ba cơ sở dữ liệu riêng biệt, cơ sở dữ liệu cho danh từ, động từ và thứ ba cho tính từ và trạng từ.

Mỗi cơ sở dữ liệu chứa một tập hợp các **lemma**, mỗi lemma được chú thích bằng một bộ các word sense. Bản phát hành **WordNet 3.0** có 117.798 danh từ, 11,529 động từ, 22.479 tính từ và 4.481 trạng từ. Danh từ trung bình có 1,23 word sense và động từ trung bình có 2,16 word sense. **WordNet** có thể được truy cập trên Web hoặc được tải xuống và truy cập cục bộ. *Hình 16,* cho thấy lemma cho danh từ và tính từ "bass".

Lưu ý rằng có 8 word sense cho danh từ và 1 cho tính từ, mỗi word sense có một **gloss** (một định nghĩa kiểu từ điển), một danh sách các từ đồng nghĩa cho word sense đó và đôi khi cũng có các ví dụ sử dụng.

Không giống như từ điển, WordNet không biểu diễn cho phát âm, do đó, không phân biệt cách phát âm.

The noun "bass" has 8 senses in WordNet.
1. bass[1] - (the lowest part of the musical range)
2. bass[2], bass part[1] - (the lowest part in polyphonic music)
3. bass[3], basso[1] - (an adult male singer with the lowest voice)
4. sea bass[1], bass[4] - (the lean flesh of a saltwater fish of the family Serranidae)
5. freshwater bass[1], bass[5] - (any of various North American freshwater fish with lean flesh (especially of the genus Micropterus))
6. bass[6], bass voice[1], basso[2] - (the lowest adult male singing voice)
7. bass[7] - (the member with the lowest range of a family of musical instruments)
8. bass[8] - (nontechnical name for any of numerous edible marine and freshwater spiny-finned fishes)

The adjective "bass" has 1 sense in WordNet.
1. bass[1], deep[6] - (having or denoting a low vocal or instrumental range)
"a deep voice"; "a bass voice is lower than a baritone voice"; "a bass clarinet"

Hình 16: Một phần WordNet 3.0 cho danh từ "bass".

Tập hợp các từ đồng nghĩa gần với word sense của WordNet được gọi là bộ từ đồng nghĩa (**synset**) (**synonym set**); **synsets** là một thứ quan trọng trong **WordNet**. Đầu vào cho "bass" bao gồm các **synset** như {bass1, deep6} hoặc {bass6, bass voice1, basso2}. Chúng ta có thể nghĩ về một **synset** đại diện cho một khái niệm về loại mà chúng ta sẽ thảo luận trong Chương 19. Do đó, thay vì biểu diễn các khái niệm theo thuật ngữ logic (term), WordNet biểu thị chúng như các danh sách các word sense có thể được sử dụng để diễn đạt khái niệm. Ở đây, một ví dụ khác về **synset**:

{chump[1], fool[2], gull[1], mark[9], patsy[1], fall guy[1], sucker[1], soft touch[1], mug[2]}

Gloss của **synset** này mô tả nó như một người dễ tin và dễ bị lợi dụng. Do đó, mỗi mục từ vựng có trong synset có thể được sử dụng để thể hiện khái niệm này.

WordNet biểu diễn cho tất cả các loại quan hệ ý nghĩa được thảo luận trong phần trước, như được minh họa trong *Hình 17* và *Hình 18*.

Relation	Also Called	Definition	Example
Hypernym	Superordinate	From concepts to superordinates	*breakfast*[1] → *meal*[1]
Hyponym	Subordinate	From concepts to subtypes	*meal*[1] → *lunch*[1]
Instance Hypernym	Instance	From instances to their concepts	*Austen*[1] → *author*[1]
Instance Hyponym	Has-Instance	From concepts to concept instances	*composer*[1] → *Bach*[1]
Member Meronym	Has-Member	From groups to their members	*faculty*[2] → *professor*[1]
Member Holonym	Member-Of	From members to their groups	*copilot*[1] → *crew*[1]
Part Meronym	Has-Part	From wholes to parts	*table*[2] → *leg*[3]
Part Holonym	Part-Of	From parts to wholes	*course*[7] → *meal*[1]
Substance Meronym		From substances to their subparts	*water*[1] → *oxygen*[1]
Substance Holonym		From parts of substances to wholes	*gin*[1] → *martini*[1]
Antonym		Semantic opposition between lemmas	*leader*[1] ⟺ *follower*[1]
Derivationally		Lemmas w/same morphological root	*destruction*[1] ⟺ *destroy*[1]

Hình 17: Quan hệ danh từ trong WordNet.

Relation	Definition	Example
Hypernym	From events to superordinate events	*fly*[9] → *travel*[5]
Troponym	From events to subordinate event (often via specific manner)	*walk*[1] → *stroll*[1]
Entails	From verbs (events) to the verbs (events) they entail	*snore*[1] → *sleep*[1]
Antonym	Semantic opposition between lemmas	*increase*[1] ⟺ *decrease*[1]
Derivationally Related Form	Lemmas with same morphological root	*destroy*[1] ⟺ *destruction*[1]

Hình 18: Quan hệ động từ trong WordNet.

3.4: Word Sense Disambiguation: Overview.

Nhiệm vụ chọn nghĩa chính xác cho một từ được gọi là định nghĩa từ (**Word Sense Disambiguation**) hoặc **WSD**. **WSD** có khả năng cải thiện nhiều tác vụ xử lý ngôn ngữ tự nhiên, bao gồm dịch máy, trả lời câu hỏi và truy xuất thông tin.

Các thuật toán **WSD** lấy đầu vào là một từ trong ngữ cảnh cùng với một kho lưu trữ cố định các word sense tiềm năng và trả về đầu ra word sense chính xác của từ đó cho việc sử dụng đó. Đầu vào và các word sense phụ thuộc vào nhiệm vụ. Đối với dịch máy từ tiếng Anh sang tiếng Tây Ban Nha, kho sense tag cho một từ tiếng Anh có thể là tập hợp các bản dịch tiếng Tây Ban Nha khác nhau…

Khi chúng ta đánh giá WSD một cách cô lập, chúng ta có thể sử dụng tập hợp các word sense từ tài nguyên từ điển như WordNet.

Rất hữu ích để phân biệt hai biến thể của tác vụ WSD chung. Trong tác vụ mẫu từ vựng (**lexical sample**), một nhóm nhỏ các từ mục tiêu được chọn trước, cùng với một kho các word sense cho mỗi từ, từ một số từ vựng. Vì tập hợp các từ và tập hợp word sense là nhỏ, nên các phương pháp học máy có giám sát thường được sử dụng để xử lý các nhiệm vụ **lexical sample**. Đối với mỗi từ, một số câu ngữ cảnh có thể được chọn và được gắn nhãn bằng tay với ý nghĩa chính xác của từ mục tiêu. Hệ thống phân loại (classifier) sau đó có thể được đào tạo với các ví dụ được dán nhãn này. Các từ mục tiêu không được gắn nhãn trong ngữ cảnh sau đó có thể được gắn nhãn bằng cách sử dụng trình phân loại được đào tạo như vậy. Công việc ban đầu trong **WSD** chỉ tập trung vào các nhiệm vụ **lexical sample** thuộc loại này, xây dựng các thuật toán dành riêng cho từ để định hướng các từ đơn lẻ như “line”, “interest”, or “plant”.

Ngược lại, trong tác vụ **all-words**, các hệ thống được cung cấp toàn bộ văn bản và từ vựng với kho word sense và được yêu cầu phân biệt từng từ trong văn bản. **All-words** task tương tự như part-of-speech tagging, ngoại trừ với một bộ thẻ lớn hơn nhiều vì mỗi lemma có một bộ riêng. Hậu quả của bộ thẻ lớn hơn này là vấn đề thưa thớt dữ liệu nghiêm trọng; không chắc là dữ liệu đào tạo đầy đủ cho mỗi từ trong bộ kiểm tra sẽ có sẵn. Hơn nữa, với số lượng từ ngữ đa nghĩa trong từ vựng, các cách tiếp cận dựa trên đào tạo một phân loại cho mỗi thuật ngữ dường như không thực tế.

Trong các phần tiếp theo, chúng ta khám phá việc áp dụng các mô hình học máy khác nhau để thực hiện định nghĩa từ ngữ (**WSD**).

3.5: Summary.

Chương này đã đề cập đến một loạt các vấn đề liên quan đến ý nghĩa của các từ vựng. Sau đây là một trong những điểm nổi bật:

- Ngữ nghĩa học từ vựng (**Lexical semantics**) là nghiên cứu về ý nghĩa của các từ và các kết nối liên quan đến ý nghĩa hệ thống giữa các từ.
- Một nghĩa của từ (**word sense**) là một khía cạnh nghĩa của từ; định nghĩa và quan hệ ý nghĩa được xác định ở cấp độ **word sense** hơn là dạng từ (wordforms).
- Đồng âm (**Homonymy**) là mối quan hệ giữa các word sense không liên quan có chung một hình thức và từ nhiều nghĩa (**polysemy**) là mối quan hệ giữa các word sense có liên quan có chung một hình thức.
- Đồng nghĩa (**Synonymy**) giữ các từ khác nhau có cùng nghĩa.
- Mối quan hệ **Hyponymy** và **hypernymy** giữ giữa các từ trong mối quan hệ phân loại.
- **WordNet** là một cơ sở dữ liệu lớn về quan hệ từ vựng cho tiếng Anh

• Định hướng từ (**WSD**) là nhiệm vụ xác định nghĩa chính xác của từ trong ngữ cảnh. Phương pháp tiếp cận được giám sát sử dụng các câu trong đó các từ riêng lẻ (**lexical sample task**) hoặc **all-words task** được gắn nhãn bằng các word sense từ một tài nguyên như WordNet. Các trình phân loại được giám sát cho WSD thường được đào tạo về các tính năng chung và bag-of-word features mô tả các từ xung quanh.

• Một đường cơ sở quan trọng đối với WSD là **most frequent sense**, tương đương, trong WordNet, là **lấy từ đầu tiên**.

• Thuật toán **Lesk** chọn word sense mà định nghĩa từ điển của nó chia sẻ nhiều từ nhất với vùng lân cận từ mục tiêu.

• Các thuật toán dựa trên biểu đồ (Graph-base) xem từ điển như là đồ thị và chọn ý nghĩa trung tâm nhất theo một cách nào đó.

• Độ tương tự của từ (**Word similarity**) có thể được tính bằng cách đo khoảng cách liên kết trong một từ điển đồng nghĩa hoặc bằng cách đo sự khác nhau giữa nội dung thông tin của hai nút.

4. Chương 18: Lexicons for Sentiment and Affect Extraction

Trong chương này, chúng ta chuyển sang các công cụ để giải thích ý nghĩa tình cảm, mở rộng nghiên cứu về phân tích tình cảm trong Chương 6. Chúng ta sử dụng từ 'cảm xúc' (**affective**), theo truyền thống trong **affective computing** (Picard, 1995) để nói về cảm xúc, tình cảm, tính cách, tâm trạng và thái độ. Ý nghĩa tình cảm có liên quan chặt chẽ đến tính chủ quan, nghiên cứu về đánh giá, ý kiến, cảm xúc và suy đoán của người nói hoặc nhà văn (Wiebe et al., 1999).

Các kiểu của trạng thái tình cảm (**affective**):

Cảm xúc (**Emotion**): Tập tương đối ngắn của phản ứng đối với việc đánh giá một sự kiện bên ngoài hoặc bên trong có ý nghĩa quan trọng. VD:(tức giận, buồn, vui, sợ hãi, xấu hổ, tự hào, phấn chấn, tuyệt vọng)

Tâm trạng (**Mood**): Khuếch tán ảnh hưởng đến trạng thái, rõ rệt nhất là sự thay đổi trong cảm giác chủ quan, cường độ thấp nhưng thời gian tương đối dài, thường không có nguyên nhân rõ ràng. VD: (vui vẻ, ảm đạm, cáu kỉnh, bơ phờ, chán nản, phấn chấn)

Lập trường giữa các cá nhân (**Interpersonal stance**): Lập trường tình cảm đối với người khác trong một tương tác cụ thể, tô màu cho sự trao đổi giữa các cá nhân trong tình huống đó. VD: (xa cách, lạnh lùng, ấm áp, hỗ trợ, khinh miệt, thân thiện)

Thái độ (**Attitude**): Tương đối tin tưởng, ảnh hưởng đến màu sắc, sở thích và các điều kiện đối với các đối tượng hoặc người. VD: (thích, yêu, ghét, định giá, ham muốn)

Đặc điểm tính cách (**Personality traits**): Cảm xúc nặng nề, khuynh hướng tính cách ổn định và xu hướng hành vi, điển hình cho một người. VD: (lo lắng, lo lắng, liều lĩnh, buồn rầu, thù địch, ghen tị)...

Chúng ta có thể thiết kế các trình trích xuất cho từng loại trạng thái tình cảm này. Chương 6 đã giới thiệu phân tích tình cảm, nhiệm vụ trích xuất hướng tích cực (positive) hoặc tiêu cực (negative) mà một nhà văn thể hiện đối với một số đối tượng. Điều này tương ứng với việc trích xuất thái độ (**Attitude**): tìm hiểu những gì mọi người thích hoặc không thích, từ đánh giá của người tiêu dùng về sách hoặc phim, bài xã luận, hoặc tình cảm công khai từ blog hoặc tweet.

Phát hiện cảm xúc (**emotion**) và tâm trạng (**mood**) rất hữu ích để phát hiện xem học sinh có bối rối, bị thu hút hoặc chắc chắn khi tương tác với hệ thống hướng dẫn hay không, liệu người gọi đến đường dây trợ giúp có bị thất vọng hay không. Chẳng hạn, phát hiện cảm xúc như sợ hãi trong tiểu thuyết, có thể giúp chúng ta theo dõi những nhóm hoặc tình huống nào đang sợ và cách điều đó thay đổi theo thời gian.

Phát hiện các quan điểm giữa các cá nhân khác nhau (**Interpersonal stance**) có thể hữu ích khi trích xuất thông tin từ các cuộc trò chuyện giữa người với người. Mục tiêu ở đây là phát hiện các lập trường như thân thiện hoặc vụng về trong các cuộc phỏng vấn hoặc các cuộc trò chuyện thân thiện, hoặc thậm chí để phát hiện tán tỉnh trong việc hẹn hò. Đối với nhiệm vụ tự động tóm tắt các cuộc họp, chúng tôi muốn có thể tự động hiểu các mối quan hệ xã hội giữa mọi người, những người thân thiện hoặc đối nghịch với ai. Một nhiệm vụ liên quan là tìm các phần của cuộc trò chuyện nơi mọi người đặc biệt hào hứng hoặc tham gia, các điểm nóng trò chuyện có thể giúp người tóm tắt tập trung vào đúng khu vực.

Phát hiện tính cách của người dùng, chẳng hạn như người dùng là người hướng ngoại hay mức độ cởi mở để trải nghiệm, có thể giúp cải thiện các tác nhân đàm thoại, có vẻ hoạt động tốt hơn nếu phù hợp với mong đợi tính cách của người dùng (Mairlie và Walker, 2008).

Trong Chương 6, chúng ta đã giới thiệu việc sử dụng phân loại Naive Bayes để phân loại tình cảm của tài liệu, một cách tiếp cận đã được áp dụng thành công cho nhiều nhiệm vụ này. Theo cách tiếp cận đó, tất cả các từ trong tập huấn luyện được sử dụng làm các tính năng để phân loại tình cảm.

Trong chương này, chúng ta tập trung vào một mô hình thay thế, trong đó thay vì sử dụng mỗi từ làm features, chúng ta chỉ tập trung vào một số từ nhất định, những từ mang tín hiệu đặc biệt mạnh mẽ đến sentiment hoặc affect. Chúng ta gọi nó là danh sách các từ tình cảm hoặc từ vựng tình cảm (**sentiment or affective lexicons**). Trong các phần tiếp theo, chúng ta giới thiệu lexicons for sentiment, thuật toán bán giám sát để tạo ra chúng và thuật toán đơn giản để sử dụng từ vựng (lexicons) để thực hiện phân tích tình cảm (sentiment analysis).

Sau đó, chúng ta chuyển sang khai thác các loại ý nghĩa tình cảm khác, bắt đầu bằng cảm xúc (emotion) và sử dụng các công cụ trực tuyến để tạo ra các từ vựng cảm xúc (emotion lexicons), sau đó tiến tới các loại ý nghĩa tình cảm khác như interpersonal stance, personality.

4.1: Available Sentiment Lexicons.

Các từ vựng cơ bản nhất gắn nhãn các từ dọc theo một chiều của biến thiên ngữ nghĩa, được gọi là "tình cảm", hoặc"định hướng ngữ nghĩa".

Trong các từ vựng đơn giản nhất, chiều này được thể hiện theo kiểu nhị phân, với một danh sách từ cho các từ tích cực (positive) và một danh sách từ cho các từ tiêu cực (negative). Lâu đời nhất là General Inquirer (Stone et al., 1966), người đã nghiên cứu sớm về tâm lý học nhận thức về nghĩa của từ (Osgood et al., 1957) và về công việc phân tích nội dung.

General Inquirer là một bộ có sẵn miễn phí với các từ vựng 1915 từ tích cực và 2291 từ tiêu cực (và cũng bao gồm các từ vựng khác mà chúng ta sẽ thảo luận trong phần tiếp theo).

The MPQA Subjectivity lexicon (Wilson et al., 2005) có 2718 từ tích cực và 4912 từ tiêu cực được rút ra từ sự kết hợp của các nguồn, bao gồm General Inquirer, đầu ra của hệ thống Hatzivassiloglou - McKeown (1997) và bootstrapped list of subjective words and phrases (Riloff và Wiebe, 2003) sau đó được dán nhãn bằng tay cho tình cảm. Mỗi cụm từ trong từ vựng cũng được dán nhãn cho độ tin cậy. The polarity lexicon của (Hu và Liu, 2004b) đưa ra 2006 từ tích cực và 4783 từ tiêu cực, được rút ra từ các đánh giá sản phẩm, được dán nhãn bằng phương pháp bootstrapping từ WordNet được mô tả trong phần tiếp theo.

Positive	admire, amazing, assure, celebration, charm, eager, enthusiastic, excellent, fancy, fantastic, frolic, graceful, happy, joy, luck, majesty, mercy, nice, patience, perfect, proud, rejoice, relief, respect, satisfactorily, sensational, super, terrific, thank, vivid, wise, wonderful, zest
Negative	abominable, anger, anxious, bad, catastrophe, cheap, complaint, condescending, deceit, defective, disappointment, embarrass, fake, fear, filthy, fool, guilt, hate, idiot, inflict, lazy, miserable, mourn, nervous, objection, pest, plot, reject, scream, silly, terrible, unfriendly, vile, wicked

Hình 19: Một số từ tích cực và tiêu cực lấy từ các bộ dữ liệu.

4.2: Semi-supervised induction of sentiment lexicons.

Một số từ vựng tình cảm được xây dựng bằng cách con người gán xếp hạng cho các từ; đây là kỹ thuật xây dựng The General Inquirer bắt đầu từ những năm 1960. Nhưng một trong những cách mạnh mẽ nhất để học từ vựng là sử dụng phương pháp học bán giám sát (semi-supervised).

Trong phần này, chúng ta giới thiệu ba phương pháp cho việc học bán giám sát rất quan trọng trong việc trích xuất từ vựng tình cảm. Cả ba phương thức đều có chung thuật toán trực quan được phác họa trong *Hình 20*:

```
function BUILDSENTIMENTLEXICON(posseeds,negseeds) returns poslex,neglex

  poslex ← posseeds
  neglex ← negseeds
  Until done
    poslex ← poslex + FINDSIMILARWORDS(poslex)
    neglex ← neglex + FINDSIMILARWORDS(neglex)
  poslex,neglex ← POSTPROCESS(poslex,neglex)
```

Hình 20: Schematic for semi-supervised sentiment lexicon

Như chúng ta sẽ thấy, các phương pháp khác nhau trong cách mà chúng sử dụng để tìm các từ có độ phân cực tương tự và trong các bước chúng sử dụng để học máy để cải thiện chất lượng của từ vựng.

4.2.1: Using seed words and adjective coordination (phối hợp tính từ).

Thuật toán Hatzivassiloglou và McKeown (1997) để dán nhãn cho tính phân cực của tính từ là cùng một kiến trúc bán giám sát được mô tả ở trên. Thuật toán của họ có bốn bước.

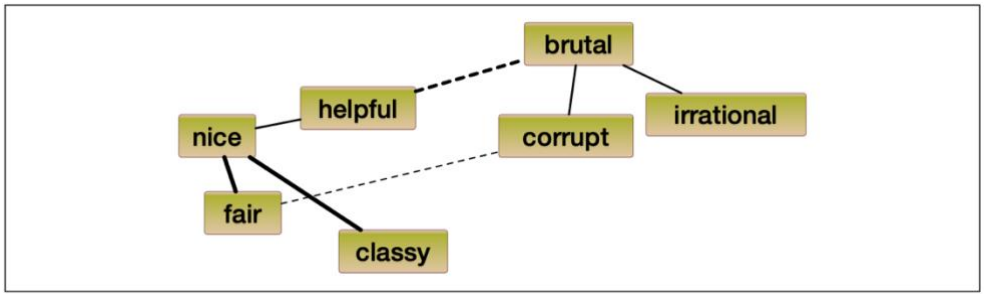
Bước 1: Tạo từ vựng hạt giống (Create seed lexicon): Đánh nhãn bằng tay một bộ hạt gồm 1336 tính từ (tất cả các từ đã xảy ra hơn 20 lần trong kho dữ liệu WSJ 21 triệu từ). Họ đã chọn 657 tính từ tích cực (ví dụ: adequate, central, clever, famous, intelligent, remarkable, reputed, sensitive, slender, thriving) và 679 tính từ tiêu cực (Ví dụ: contagious, drunken, ignorant, lanky, listless, primitive, strident, troublesome, unre- solved, unsuspecting).

Bước 2: Tìm tín hiệu cho các từ ứng cử viên tương tự: Chọn các từ giống hoặc khác với các từ seed word, sử dụng trực giác mà các tính từ liên kết bởi các từ "and" có xu hướng có cùng cực. Do đó, chúng ta có thể mong đợi thấy các trường hợp tính từ tích cực phối hợp với tích cực hoặc tiêu cực với tiêu cực. Ngược lại, tính từ nối bằng "but" có khả năng có cực ngược nhau.

Ý tưởng rằng các mẫu đơn giản như phối hợp thông qua từ "and" là công cụ tốt để tìm các mối quan hệ từ vựng như cùng cực và phân cực ngược là một ứng dụng của phương pháp dựa trên mẫu để trích xuất quan hệ được mô tả trong Chương 20.

Một số gợi ý khác cho sự phân cực đối nghịch xuất phát từ sự phủ định hình thái (un-, im-, -less). Tính từ có cùng một gốc nhưng khác nhau về một hình thái âm có xu hướng trái ngược nhau.

Bước 3: Build a polarity graph: Các tín hiệu này được tích hợp bằng cách xây dựng một đồ thị với các nút cho các từ và liên kết thể hiện khả năng hai từ có cùng cực như thế nào, như trong *Hình 21*.

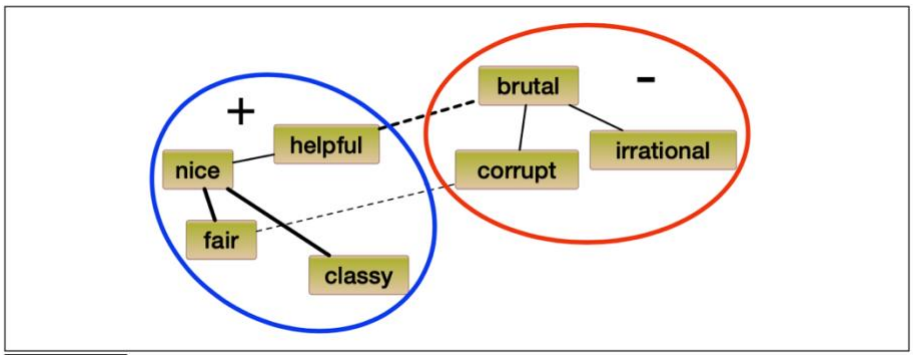


Hình 21: Đồ thị polarity similarity giữa các cặp từ.

Một cách đơn giản để xây dựng đồ thị sẽ dự đoán một liên kết phân cực ngược nếu hai tính từ được kết nối bởi ít nhất một từ "but" và một liên cùng cực (đối với bất kỳ hai tính từ nào được kết nối bởi ít nhất một liên kết conjunction). Phương pháp phức tạp hơn được sử dụng bởi Hatzivassiloglou và McKeown (1997) là xây dựng một bộ phân loại có giám sát để dự đoán liệu hai từ có cùng cực hay khác nhau hay không, bằng cách sử dụng 3 features (xuất hiện với "and", xảy ra với "but", và phủ định hình thái).

Trình phân loại được đào tạo trên một tập hợp con của các seed word được dán nhãn bằng tay và trả về một xác suất rằng mỗi cặp từ có cùng cực hoặc đối cực. "polarity similarity" này của mỗi cặp từ có thể được xem là sức mạnh của các liên kết tích cực hoặc tiêu cực giữa chúng trong một biểu đồ.

Bước 4: Clustering the graph: Cuối cùng, bất kỳ thuật toán phân cụm biểu đồ có thể được sử dụng để phân chia biểu đồ thành hai tập hợp con có cùng cực; một minh hoạ được hiển thị trong *Hình 22*.



Hình 22:Clustering the graph.

4.2.2: Pointwise mutual information (PMI)

Phương thức đầu tiên để tìm các từ có độ phân cực tương tự dựa trên các mẫu kết hợp, bây giờ chúng ta chuyển sang phương thức thứ hai sử dụng sự xuất hiện lân cận làm đại diện cho độ polarity similarity. Thuật toán này giả định rằng các từ có cực tính tương tự có xu hướng xảy ra gần nhau, sử dụng thuật toán Pointwise mutual information (PMI) được định nghĩa trong Chương 15.

Phương pháp của Turney (2002) sử dụng phương pháp này để gán cực cho cả từ và cụm từ có hai từ.

Trong bước trước, các cụm từ hai từ được trích xuất dựa trên các biểu thức chính quy đơn giản của lời nói. Các biểu thức chọn danh từ có tính từ đứng trước, động từ có trạng từ đứng trước,…

Để đo tính phân cực của từng cụm từ được trích xuất, chúng ta bắt đầu bằng cách chọn các từ hạt giống tích cực và tiêu cực. Ví dụ, chúng ta có thể chọn một từ hạt giống tích cực duy nhất "excellent" và một từ hạt giống tiêu cực duy nhất "poor". Sau đó, chúng ta sử dụng trực giác rằng các cụm từ tích cực nói chung sẽ có xu hướng cùng xuất hiện nhiều hơn với "excellent". Cụm từ tiêu cực đồng xảy ra nhiều với "poor".

Biện pháp **PMI** có thể được sử dụng để đo lường sự xuất hiện này. Nhớ lại từ Chương 15 rằng **PMI** (Fano, 1961) là thước đo mức độ thường xuyên xảy ra hai sự kiện x và y, so với những gì chúng ta mong đợi, nếu chúng độc lập:

$$PMI(x,y) = \log_2 \frac{P(x,y)}{P(x)P(y)}$$

Trực giác này có thể được áp dụng để đo lường sự xuất hiện của hai từ bằng cách xác định PMI giữa một từ hạt giống s và một từ khác w:

$$\text{PMI}(w,s) = \log_2 \frac{P(w,s)}{P(w)P(s)}$$

Turney (2002) ước tính xác suất của phương trình trên bằng cách sử dụng công cụ tìm kiếm với toán tử NEAR, chỉ định rằng một từ phải ở gần (near) một từ khác. Các xác suất sau đó được ước tính như sau:

$$P(w) = \frac{\text{hits}(w)}{N}$$
$$P(w1,w2) = \frac{\text{hits}(w1 \text{ NEAR } w2)}{kN}$$

Đó là, chúng ta ước tính xác suất của một từ là số đếm được trả về từ công cụ tìm kiếm, được chuẩn hóa bằng tổng số từ trong toàn bộ trang web (N). (Không quan trọng là chúng ta không biết N là gì, vì hóa ra nó sẽ hủy bỏ độc đáo). Xác suất bigram là số lần xuất hiện bigram được normalization mặc dù có N unigram và cũng có khoảng N bigram trong một kho có độ dài N, có kN "NEAR" bigram, trong đó hai từ được phân tách bằng khoảng cách lên đến k.

PMI giữa hai từ w và s là:

$$\text{PMI}(w,s) = \log_2 \frac{\frac{1}{kN}\text{hits}(w \text{ NEAR } s)}{\frac{1}{N}\text{hits}(w)\frac{1}{N}\text{hits}(s)}$$

Cái nhìn sâu sắc của Turney (2002) là để xác định tính phân cực của một từ bằng cách nó xuất hiện bao nhiêu với các hạt giống tích cực và không xảy ra với các hạt giống tiêu cực:

$$\begin{aligned}
\text{Polarity}(w) &= \text{PMI}(w, \text{"excellent"}) - \text{PMI}(w, \text{"poor"}) \\
&= \log_2 \frac{\frac{1}{kN}\text{hits}(w \text{ NEAR "excellent"})}{\frac{1}{N}\text{hits}(w)\frac{1}{N}\text{hits}(\text{"excellent"})} - \log_2 \frac{\frac{1}{kN}\text{hits}(w \text{ NEAR "poor"})}{\frac{1}{N}\text{hits}(w)\frac{1}{N}\text{hits}(\text{"poor"})} \\
&= \log_2 \left(\frac{\text{hits}(w \text{ NEAR "excellent"})}{\text{hits}(w)\text{hits}(\text{"excellent"})} \frac{\text{hits}(w)\text{hits}(\text{"poor"})}{\text{hits}(w \text{ NEAR "poor"})} \right) \\
&= \log_2 \left(\frac{\text{hits}(w \text{ NEAR "excellent"})\ \text{hits}(\text{"poor"})}{\text{hits}(\text{"excellent"})\ \text{hits}(w \text{ NEAR "poor"})} \right) \qquad (18.6)
\end{aligned}$$

Bảng dưới đây từ Turney (2002) cho thấy các ví dụ mẫu về các cụm từ được học theo phương pháp PMI:

Extracted Phrase	Polarity
online experience	2.3
very handy	1.4
low fees	0.3
inconveniently located	-1.5
other problems	-2.8
unethical practices	-8.5

4.2.3: Using WordNet synonyms and antonyms.

Phương pháp thứ ba để tìm các từ có similar polarity với các seed word là sử dụng từ đồng nghĩa và từ trái nghĩa. Trực giác là các từ đồng nghĩa của một từ có thể chia sẻ tính phân cực (polarity) của nó trong khi một từ trái nghĩa của một từ có thể ngược lại.

Vì WordNet có các mối quan hệ này, nó thường được sử dụng (Kim và Hovy 2004, Hu và Liu 2004b). Sau khi một từ vựng hạt giống được xây dựng, mỗi từ vựng được cập nhật như sau, có thể được lặp lại.

Lex +: Thêm từ đồng nghĩa của từ tích cực ("well") và từ trái nghĩa của từ tiêu cực.

Lex -: Thêm từ đồng nghĩa của từ tiêu cực ("awful") và từ trái nghĩa của từ tích cực.

Một phần mở rộng của thuật toán này đã được áp dụng để gán phân cực cho các word sense của WordNet, được gọi là SentiWordNet (Baccianella et al., 2010). *Hình 23* cho thấy một số ví dụ.

Synset	Pos	Neg	Obj
good#6 'agreeable or pleasing'	1	0	0
respectable#2 honorable#4 good#4 estimable#2 'deserving of esteem'	0.75	0	0.25
estimable#3 computable#1 'may be computed or estimated'	0	0	1
sting#1 burn#4 bite#2 'cause a sharp or stinging pain'	0	0.875	.125
acute#6 'of critical importance and consequence'	0.625	0.125	.250
acute#4 'of an angle; less than 90 degrees'	0	0	1
acute#1 'having or experiencing a rapid onset and short but severe course'	0	0.5	0.5

Hình 23: Ví dụ từ SentiWordNet 3.0

Trong thuật toán này, tính phân cực được gán cho toàn bộ synset (tập hợp các từ đồng nghĩa) thay vì một từ. Một từ vựng tích cực được xây dựng từ tất cả các từ đồng nghĩa liên quan đến 7 từ tích cực và từ vựng tiêu cực từ các từ đồng nghĩa liên quan đến 7 từ tiêu cực. Cả hai đều được mở rộng bằng cách vẽ trong các từ đồng bộ liên quan đến các mối quan hệ WordNet như từ trái nghĩa hoặc nhìn thấy. Sau đó, một bộ phân loại được đào tạo từ dữ liệu này để lấy gloss (nét ý nghĩa của synset) WordNet và quyết định xem ý nghĩa được xác định là tích cực, tiêu cực hay trung tính. Một bước nữa (liên quan đến thuật toán randomwalk) gán điểm cho mỗi synset WordNet cho mức độ tích cực, tiêu cực và tính trung lập của nó.

Tóm lại, chúng ta đã thấy ba cách khác nhau để sử dụng học tập bán giám sát để tạo ra một từ vựng tình cảm. Tất cả bắt đầu bằng một tập hợp các từ hạt giống tích cực và tiêu cực, nhỏ như 2 từ (Turney, 2002) hoặc lớn như một nghìn từ (Hatzivassiloglou và McKeown, 1997). Sau đó, nhiều từ có độ phân cực tương tự được thêm vào, sử dụng các phương thức dựa trên pattern, sự xuất hiện của PMI hoặc từ đồng nghĩa và từ trái nghĩa của WordNet. Classifier cũng có thể được sử dụng để kết hợp các tín hiệu khác nhau để phân cực của các từ mới, bằng cách đào tạo trên các bộ huấn luyện hạt giống, hoặc lặp lại sớm.

4.3: Supervised learning of word sentiment.

Phần trước cho thấy các cách bán giám sát để tìm hiểu tình cảm khi không có tín hiệu giám sát, bằng cách mở rộng một bộ hạt giống được xây dựng bằng tay tìm polarity similarity. Một cách khác là học tập có giám sát, sử dụng trực tiếp một nguồn giám sát mạnh mẽ cho word sentiment: VD: đánh giá trực tuyến (on-line reviews).

Trang web chứa một số lượng lớn các đánh giá trực tuyến cho nhà hàng, phim, sách hoặc các sản phẩm khác, mỗi sản phẩm có nội dung đánh giá cùng với điểm đánh giá liên quan: giá trị có thể dao động từ 1 sao đến 5 sao, hoặc chấm điểm từ 1 đến 10. *Hình 24* cho thấy các mẫu được trích xuất từ các đánh giá về nhà hàng, sách và phim.

	Movie review excerpts (IMDB)
10	A great movie. This film is just a wonderful experience. It's surreal, zany, witty and slapstick all at the same time. And terrific performances too.
1	This was probably the worst movie I have ever seen. The story went nowhere even though they could have done some interesting stuff with it.
	Restaurant review excerpts (Yelp)
5	The service was impeccable. The food was cooked and seasoned perfectly... The watermelon was perfectly square ... The grilled octopus was ... mouthwatering...
2	...it took a while to get our waters, we got our entree before our starter, and we never received silverware or napkins until we requested them...
	Book review excerpts (GoodReads)
1	I am going to try and stop being deceived by eye-catching titles. I so wanted to like this book and was so disappointed by it.
5	This book is hilarious. I would recommend it to anyone looking for a satirical read with a romantic twist and a narrator that keeps butting in
	Product review excerpts (Amazon)
5	The lid on this blender though is probably what I like the best about it... enables you to pour into something without even taking the lid off! ... the perfect pitcher! ... works fantastic.
1	I hate this blender... It is nearly impossible to get frozen fruit and ice to turn into a smoothie... You have to add a TON of liquid. I also wish it had a spout ...

Hình 24: Trích từ một số đánh giá từ các trang web khác nhau.

Chúng ta có thể sử dụng điểm đánh giá này làm giám sát: các từ tích cực có nhiều khả năng xuất hiện trong các đánh giá 5 sao; từ tiêu cực trong đánh giá 1 sao. Và thay vì chỉ là một cực nhị phân, loại giám sát này cho phép chúng ta gán cho một từ đại diện phức tạp hơn cho cực của nó: phân phối của nó trên các số sao (hoặc các điểm khác).

Do đó, trong một hệ thống mười sao, chúng ta có thể biểu thị tình cảm của mỗi từ dưới dạng 10-tuple, mỗi số là một điểm số đại diện cho sự liên kết của từ đó với mức độ phân cực đó. Liên kết này có thể là số đếm thô hoặc likelihood P (c|w) hoặc một số hàm khác của số đếm, cho mỗi lớp c từ 1 đến 10.

Ví dụ: chúng ta có thể tính toán IMDB likelihood của một từ như "disapoint" (ed / ing) xảy ra trong đánh giá 1 sao bằng cách chia số lần "disapoint" (ed / ing) xảy ra trong các đánh giá 1 sao trong bộ dữ liệu IMDB (8,557) cho tổng số từ xuất hiện trong các đánh giá 1 sao (25.395.214), do đó, ước tính IMDB của P ("disapoint"| 1) là 0,0003.

Một sửa đổi nhỏ của trọng số này, normalized likelihood, có thể được sử dụng (Potts, 2011):

$$P(w|c) = \frac{count(w,c)}{\sum_{w \in C} count(w,c)}$$

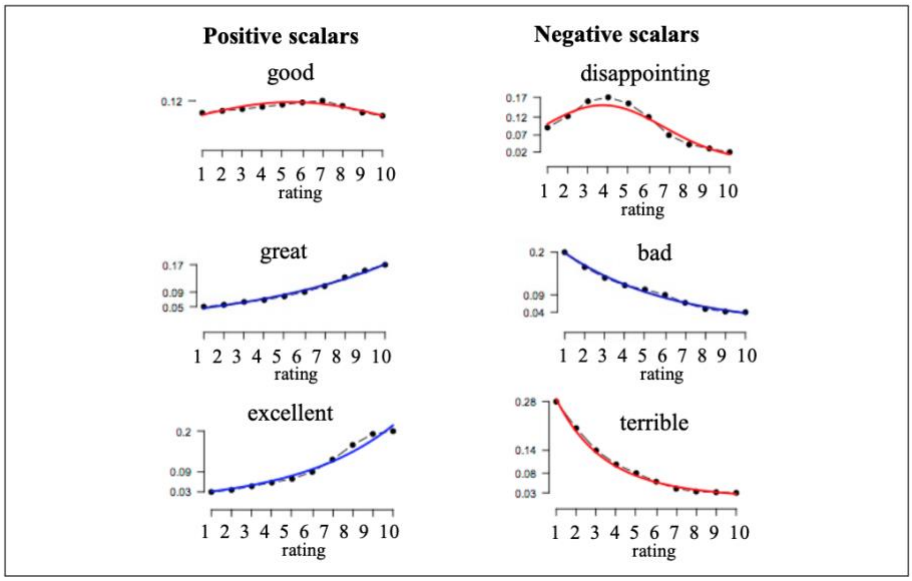
$$PottsScore(w) = \frac{P(w|c)}{\sum_{c} P(w|c)}$$

Chia ước tính IMDB P("disapoint" | 1) là 0,0003 cho tổng likelihood P (w|c) trên tất cả các danh mục (categories) cho điểm Potts là 0,10. Do đó, từ "disappoint" được liên kết với vectơ [.10, .12, .14, .14, .13, .11, .08, .06, .06, .05]. Biểu đồ Potts (Potts, 2011) là một hình ảnh trực quan của các điểm số từ này, đại diện cho tình cảm trước đó của một từ dưới dạng phân phối trên các xếp hạng (categories).

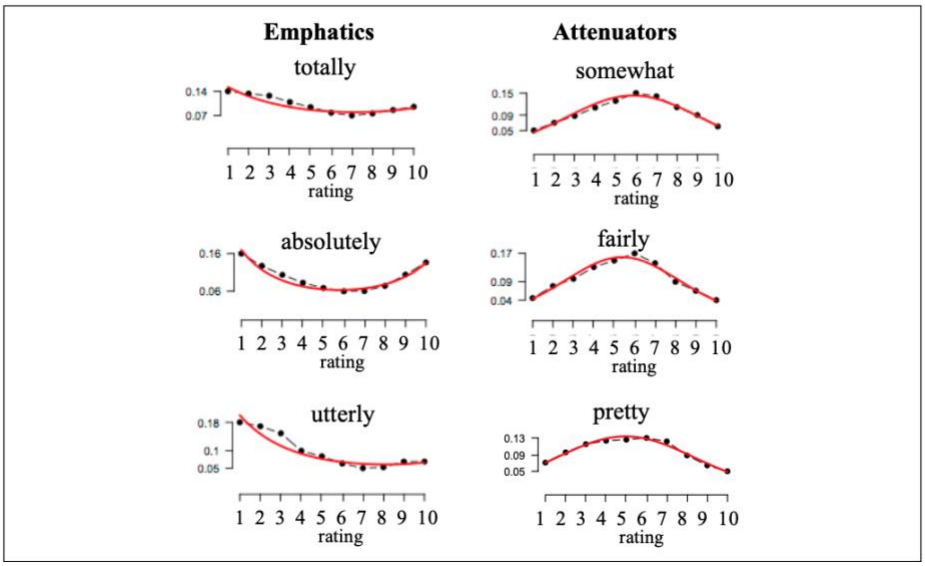
Hình 25 cho thấy sơ đồ Potts cho 3 tính từ tích cực và 3 tính từ tiêu cực. Những hình dạng này cung cấp một cách nhìn của ý nghĩa từ tình cảm.

Hình 26 cho thấy các sơ đồ Potts để nhấn mạnh và làm giảm các trạng từ. Một lần nữa chúng ta thấy sự khái quát trong các đường cong đặc trưng liên quan đến các từ có ý nghĩa cụ thể.

Các sơ đồ có thể được sử dụng như một kiểu của lexical sentiment, và cũng đóng một vai trò trong việc mô hình hóa thành phần tình cảm.



Hình 25: Potts diagrams (Potts, 2011) cho tính từ.



Hình 26: Potts diagrams (Potts, 2011) cho trạng từ.

4.3.1: Log odds ratio informative Dirichlet prior.

Một điều chúng ta thường muốn làm với phân cực từ là phân biệt giữa các từ có nhiều khả năng được sử dụng trong một loại văn bản hơn là trong một loại khác. Ví dụ, chúng ta có thể muốn biết các từ được liên kết nhiều nhất với các đánh giá 1 sao so với các từ được liên kết với các đánh giá 5 sao. Những khác biệt này có thể không chỉ liên quan đến tình cảm. Chúng ta có thể muốn tìm những từ được Đảng Dân chủ sử dụng thường xuyên hơn các thành viên của Đảng Cộng hòa, hoặc những từ được sử dụng thường xuyên hơn trong thực đơn của các nhà hàng đắt tiền hơn là các nhà hàng giá rẻ.

Đưa ra hai loại tài liệu, để tìm các từ liên quan đến một loại hơn một loại khác, chúng ta có thể chọn chỉ tính toán sự khác biệt về tần số (là một từ w thường xuyên hơn trong lớp A hoặc lớp B?). Hoặc thay vì sự khác biệt về tần số, chúng tôi có thể muốn tính tỷ lệ tần số hoặc log-odds ratio (log của tỷ lệ chênh lệch của hai từ). Sau đó, chúng ta có thể sắp xếp các từ

theo bất kỳ liên kết nào với danh mục chúng ta sử dụng, (sắp xếp từ các từ được mô tả trong danh mục A đến các từ được mô tả trong danh mục B).

Nhiều số liệu như vậy đã được nghiên cứu; trong phần này chúng ta sẽ tìm hiểu chi tiết về một trong số đó, phương pháp "log odds ratio informative Dirichlet prior" của Monroe et al. (2008) đó là một phương pháp đặc biệt hữu ích để tìm các từ được thống kê quá mức trong một loại văn bản cụ thể so với loại khác.

Phương pháp ước tính sự khác biệt giữa tần số của từ w trong hai khối văn bản i và j thông log-odds-ratio cho w, $\delta_w^{(i-j)}$, được ước tính là:

$$\delta_w^{(i-j)} = \log\left(\frac{y_w^i + \alpha_w}{n^i + \alpha_0 - (y_w^i + \alpha_w)}\right) - \log\left(\frac{y_w^j + \alpha_w}{n^j + \alpha_0 - (y_w^j + \alpha_w)}\right)$$

Trong đó ni là kích thước của corpus i, nj là kích thước của corpus j, ywi là số lượng từ w trong corpus i, ywj là số lượng từ w trong corpus j, α0 là kích thước của background corpus và αw là số lượng từ w trong background corpus.)

Ngoài ra, Monroe và cộng sự. (2008) sử dụng ước tính cho variance of the log–odds–ratio:

$$\sigma^2\left(\hat{\delta}_w^{(i-j)}\right) \approx \frac{1}{y_w^i + \alpha_w} + \frac{1}{y_w^j + \alpha_w}$$

Thống kê cuối cùng cho một từ sau đó là điểm z-score của log–odds–ratio:

$$\frac{\hat{\delta}_w^{(i-j)}}{\sqrt{\sigma^2\left(\hat{\delta}_w^{(i-j)}\right)}}$$

Monroe và cộng sự (2008) sửa log-odds ratio thường được sử dụng theo hai cách: nó sử dụng z-score của log-odds ratio, điều khiển lượng phương sai trong tần số từ và sử dụng tổng số đếm (count) từ một background corpus để cung cấp số đếm trước cho các từ, về cơ bản chia nhỏ số đếm về prior frequency trong background corpus lớn.

4.4: Using Lexicons for Sentiment Recognition.

Trong Chương 6, chúng ta đã giới thiệu thuật toán Bayes ngây thơ để phân tích tình cảm. Các từ vựng chúng ta đã tập trung trong suốt chương cho đến nay có thể được sử dụng theo một số cách để cải thiện phát hiện tình cảm.

Trong trường hợp đơn giản nhất, từ vựng có thể được sử dụng khi chúng ta không có đủ dữ liệu đào tạo để xây dựng một bộ phân tích tình cảm được giám sát; thường có thể tốn kém khi có một con người gán nhãn cho mỗi tài liệu để đào tạo bộ phân loại được giám sát.

Trong các tình huống như vậy, từ vựng có thể được sử dụng trong một thuật toán dựa trên quy tắc đơn giản để phân loại. Phiên bản đơn giản nhất chỉ là sử dụng tỷ lệ giữa các từ tích cực với các từ phủ định: nếu một tài liệu có nhiều từ tích cực hơn các từ phủ định (sử dụng từ vựng để quyết định độ phân cực của mỗi từ trong tài liệu), thì nó được phân loại là tích cực. Thường thì một ngưỡng được sử dụng, trong đó một tài liệu được phân loại là tích cực chỉ khi tỷ lệ này lớn hơn ngưỡng. Nếu từ vựng tình cảm bao gồm các trọng số dương và âm cho mỗi từ, θw + và θw−, chúng cũng có thể được sử dụng. Ở đây, một thuật toán sentiment đơn giản như vậy:

$$f^+ = \sum_{w \text{ s.t. } w \in positivelexicon} \theta_w^+ count(w)$$

$$f^- = \sum_{w \text{ s.t. } w \in negativelexicon} \theta_w^- count(w)$$

$$sentiment = \begin{cases} + & \text{if } \frac{f^+}{f^-} > \lambda \\ - & \text{if } \frac{f^-}{f^+} > \lambda \\ 0 & \text{otherwise.} \end{cases}$$

Nếu dữ liệu đào tạo được giám sát có sẵn, các số đếm này được tính toán từ các từ vựng tình cảm, đôi khi có trọng số hoặc được chuẩn hóa theo nhiều cách khác nhau, cũng có thể được sử dụng làm các tính năng trong phân loại cùng với các tính năng từ vựng hoặc phi từ vựng khác.

4.5: Emotion and other classes.

Một trong những lớp tình cảm quan trọng nhất là cảm xúc (emotion), mà Scherer (2000) định nghĩa là một tập phản ứng tương đối ngắn gọn đối với việc đánh giá một sự kiện bên ngoài hoặc bên trong là có ý nghĩa quan trọng.

Phát hiện cảm xúc có khả năng cải thiện một số nhiệm vụ xử lý ngôn ngữ. Tự động phát hiện cảm xúc trong đánh giá hoặc phản hồi của khách hàng (tức giận, không hài lòng, tin tưởng) có thể giúp doanh nghiệp nhận ra các vấn đề cụ thể hoặc các vấn đề đang diễn ra tốt đẹp. Nhận dạng cảm xúc có thể giúp các hệ thống hộp thoại như hệ thống dạy kèm phát hiện ra rằng một học sinh không vui, buồn chán, do dự, tự tin, v.v. Cảm xúc có thể đóng một vai trò trong các nhiệm vụ tin học y tế như phát hiện trầm cảm hoặc ý định tự tử.

Có hai nhóm lý thuyết về cảm xúc. Trong một nhóm, cảm xúc được xem là đơn vị nguyên tử cố định, bị giới hạn về số lượng và từ đó tạo ra những cảm xúc khác, thường được gọi là cảm xúc cơ bản (Tomkins 1962, Plutchik 1962). Có lẽ nổi tiếng nhất trong nhóm lý thuyết này là 6 cảm xúc được đề xuất bởi (Ekman, 1999) như một tập hợp những cảm xúc có khả năng phổ biến trong tất cả các nền văn hóa: “surprise, happiness, anger, fear, disgust, sadness”. Một lý thuyết nguyên tử khác là bánh xe cảm xúc (Plutchik, 1980), bao gồm 8 cảm xúc cơ bản trong bốn cặp đối lập: “joy–sadness, anger–fear, trust–disgust, và anticipation–surprise”, cùng với những cảm xúc xuất phát từ chúng, được thể hiện trong hình 25.

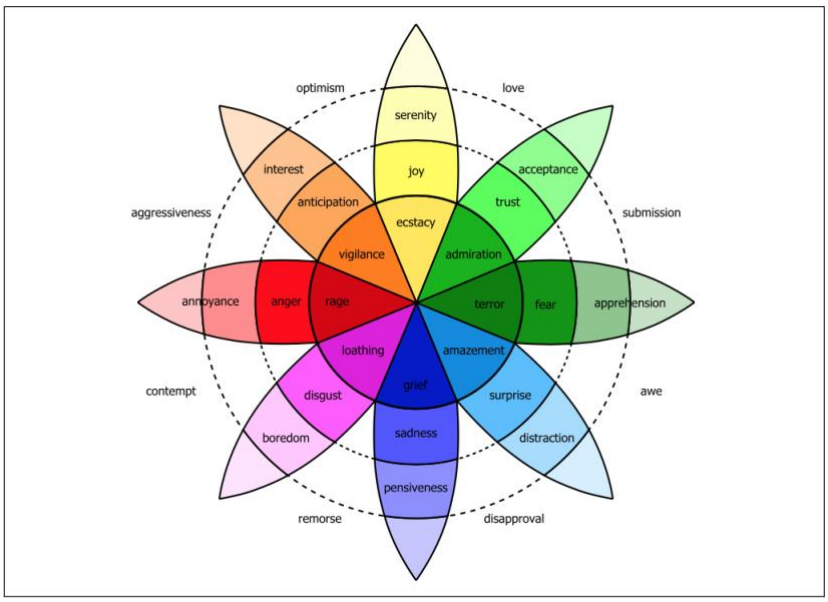
Lớp lý thuyết cảm xúc thứ hai xem cảm xúc như một không gian trong 2 hoặc 3 chiều (Russell, 1980). Hầu hết các mô hình bao gồm hai chiều **valence** và **arousal**, và thêm chiều thứ ba **dominance**. Những thứ này có thể được định nghĩa là:

valence: sự dễ chịu của kích thích

arousal: cường độ của cảm xúc bị kích thích bởi sự kích thích

dominance: mức độ kiểm soát của tác nhân kích thích

Từ vựng thực tế đã được xây dựng cho cả hai loại lý thuyết về cảm xúc.



Hình 27: Plutchik wheel of emotion.

4.5.1: Lexicons for emotion and other affective states.

Trong khi các thuật toán bán giám sát là chuẩn mực trong sentiment và polarity, cách phổ biến nhất để xây dựng emotional lexicons là để con người gắn nhãn các từ. Điều này thường được thực hiện bằng cách sử dụng dịch vụ đám đông (**crowdsourcing**): chia nhỏ nhiệm vụ thành các phần nhỏ và phân phối chúng cho một số lượng lớn các người chú thích. Chúng ta hãy xem xét một crowdsourced emotion lexicon cho mỗi trong hai mô hình lý thuyết phổ biến về emotion.

NRC Word-Emotion Association Lexicon, còn được gọi là **EmoLex** (Mohammad và Turney, 2013), sử dụng 8 cảm xúc cơ bản của Plutchik (1980) được định nghĩa ở trên. Từ vựng bao gồm khoảng 14.000 từ được chọn một phần từ các từ vựng trước đó (the General Inquirer and WordNet Affect Lexicons) và một phần từ Macquarie Thesaurus, trong đó 200 từ thường xuyên nhất được chọn từ bốn phần của lời nói: danh từ, động từ, trạng từ và tính từ (sử dụng tần số từ Google n-gram count).

Để đảm bảo rằng các chú thích đang đánh giá đúng nghĩa của từ này, trước tiên họ đã trả lời một câu hỏi đồng nghĩa nhiều lựa chọn, đưa ra ý nghĩa chính xác của từ này (không yêu cầu người chú thích đọc một định nghĩa có nghĩa khó hiểu). Một ví dụ:

```
Which word is closest in meaning (most related) to startle?
```

- automobile
- shake
- honesty
- entertain

Lớp lý thuyết cảm xúc thứ hai, cũng được xây dựng bằng cách sử dụng **crowdsourcing**, gán các giá trị trên ba chiều (valence/arousal/dominance) cho 14.000 từ (Warriner et al., 2013).

Các chú thích được đánh dấu mỗi từ có giá trị từ 1-9 trên mỗi chiều, với tỷ lệ được xác định cho chúng như sau:

- **valence**
  - 9: happy, pleased, satisfied, contented, hopeful
  - 1: unhappy, annoyed, unsatisfied, melancholic, despaired, or bored
- **arousal**
  - 9: stimulated, excited, frenzied, jittery, wide-awake, or aroused
  - 1: relaxed, calm, sluggish, dull, sleepy, or unaroused;
- **dominance**
  - 9: in control, influential, important, dominant, autonomous, or controlling
  - 1: controlled, influenced, cared-for, awed, submissive, or guided

Valence		Arousal		Dominance	
vacation	8.53	rampage	7.56	self	7.74
happy	8.47	tornado	7.45	incredible	7.74
whistle	5.7	zucchini	4.18	skillet	5.33
conscious	5.53	dressy	4.15	concur	5.29
torture	1.4	dull	1.67	earthquake	2.14

4.6: Other tasks: Personality (Tính cách cá nhân).

Nhiều loại ý nghĩa tình cảm khác có thể được trích xuất từ văn bản và lời nói. Ví dụ: phát hiện tính cách của một người từ ngôn ngữ của họ có thể hữu ích cho các hệ thống tin nhắn (người dùng có xu hướng thích các sản phẩm phù hợp với tính cách của họ) và có thể đóng vai trò hữu ích trong các câu hỏi khoa học xã hội tính toán như hiểu cách tính cách liên quan đến các loại hành vi khác.

Nhiều giả thuyết về tính cách con người dựa trên một số lượng kích thước nhỏ, chẳng hạn như các phiên bản khác nhau của "Big Five" (Digman, 1990):

Extroversion vs. Introversion (Hướng ngoại với hướng nội): sociable, assertive, playful vs. aloof, reserved, shy

Emotional stability vs. Neuroticism (Cảm xúc ổn định với hỗn loạn): calm, unemotional vs. insecure, anxious

Agreeableness vs. Disagreeableness (Dễ chịu với khó chịu): friendly, cooperative vs. antagonistic, faultfinding

Conscientiousness vs. Unconscientiousness (Tận tâm với tắc trách): self-disciplined, organized vs. inefficient, careless

Openness to experience (Cởi mở để trải nghiệm): intellectual, insightful vs. shallow, unimaginative

Một vài tập văn bản và bài phát biểu đã được dán nhãn cho tính cách của họ bằng cách các tác giả làm bài kiểm tra tính cách tiêu chuẩn. Tập tiểu luận của Pennebaker and King (1999) bao gồm 2.479 bài luận (1,9 triệu từ) từ các sinh viên tâm lý học được yêu cầu "viết bất cứ điều gì trong tâm trí bạn" trong 20 phút. The EAR (Electronically Activated Recorder) của Mehl et al. (2006) được tạo ra bằng cách các tình nguyện viên đeo máy ghi âm suốt cả ngày, trong đó ghi lại ngẫu nhiên các đoạn hội thoại ngắn trong suốt cả ngày, sau đó được phiên âm. Facebook corpus của (Schwartz et al., 2013) bao gồm 309 triệu từ của các bài đăng trên Facebook từ 75.000 tình nguyện viên.

4.7: Affect Recognition.

Phát hiện cảm xúc, tính cách, lập trường và các loại ý nghĩa tình cảm khác được mô tả bởi Scherer (2000) có thể được thực hiện bằng cách khái quát các thuật toán được mô tả ở trên để phát hiện tình cảm.

Các thuật toán phổ biến nhất liên quan đến phân loại có giám sát: một tập huấn luyện được dán nhãn ý nghĩa tình cảm, và một trình phân loại được xây dựng bằng các features được trích xuất từ tập huấn luyện. Như với phân tích tình cảm (sentiment), nếu tập huấn luyện đủ lớn và tập kiểm tra đủ tương tự với tập huấn luyện, chỉ cần sử dụng tất cả các từ hoặc tất cả các bigram như các features trong bộ phân loại mạnh như SVM hoặc hồi quy logistic, là một thuật toán xuất sắc có hiệu năng khó đánh bại. Vì vậy, chúng ta có thể coi phân loại ý nghĩa tình cảm của một mẫu văn bản là phân loại tài liệu đơn giản.

4.8: Summary.

• Nhiều loại trạng thái tình cảm (**affective**) có thể được phân biệt, bao gồm cảm xúc (emotions), tâm trạng (moods), thái độ (attitudes) (bao gồm cả tình cảm), lập trường giữa các cá nhân (interpersonal stance) và tính cách (personality).

• Các từ có các khía cạnh ý nghĩa liên quan đến các trạng thái tình cảm này, và khía cạnh ý nghĩa này của từ có thể được biểu diễn trong từ vựng (**lexicons**).

• Từ vựng cảm xúc (**Affective lexicons**) có thể được xây dựng bằng tay, sử dụng nguồn **crowd sourcing** để gắn nhãn nội dung cảm xúc của mỗi từ.

• Lexicons có thể được xây dựng bán giám sát, bootstrapping từ các từ hạt giống (**seed words**) bằng cách sử dụng các số liệu tương tự (similarity metrics) như tần số hai từ được kết hợp bởi “and” hoặc “but”, PMI của hai từ hoặc liên kết của chúng thông qua quan hệ từ đồng nghĩa hoặc trái nghĩa của WordNet.

• Lexicons có thể được học theo cách được giám sát đầy đủ, khi có thể tìm thấy dữ liệu đào tạo thuận tiện trên thế giới, chẳng hạn như xếp hạng được đánh giá bởi người dùng trên trang web.

• Các từ có thể được gán trọng số trong một từ vựng bằng cách sử dụng các hàm khác nhau của số lượng từ trong văn bản đào tạo và các số liệu tỷ lệ như **log odds ratio informative Dirichlet prior**

• Cảm xúc (**Emotion**) có thể được biểu thị bằng các đơn vị nguyên tử cố định thường được gọi là cảm xúc cơ bản, hoặc là các điểm trong không gian được xác định bởi các chiều như “**valence**” và “**arousal**”.

• Tính cách (**personality**) thường được thể hiện như một điểm trong không gian 5 chiều.

• Cảm xúc (**affective**) có thể được phát hiện, giống như tình cảm (**sentiment**), bằng cách sử dụng các kỹ thuật phân loại văn bản được giám sát tiêu chuẩn, sử dụng tất cả các từ hoặc bigram trong văn bản làm features. Các tính năng bổ sung có thể được rút ra từ số lượng từ trong từ vựng (lexicons).

• Lexicons cũng có thể được sử dụng để phát hiện cảm xúc trong phân loại dựa trên quy tắc (**rule-based classifier**) bằng cách chọn tình cảm đa số đơn giản dựa trên số lượng từ trong mỗi từ vựng.

DANH MỤC CÁC TÀI LIỆU THAM KHẢO VÀ THAM CHIẾU

[1] Daniel Jurafsky and James H. Martin. "Speech and Language Processing", 3nd Edition