

Nguyễn Hoàng Dương (B19DCCN153) 41:18

COSINE SIMILARITY

COSINE SIMILARITY

Ứng dụng đo độ tương tự hai vector nhận diện mỗi tấm ảnh đưa vào là khuôn mặt của ai (danh tính) bằng công thức toán tính góc của hai vector:

$$\text{similarity} = \cos \theta = \frac{\vec{A} \cdot \vec{B}}{|\vec{A}| |\vec{B}|}$$

Trong bài toán xử lý ảnh, đầu ra của giải thuật nhận diện khuôn mặt là 1 vector đặc trưng. Mỗi ảnh trong thư viện ảnh đã có và ảnh cần nhận diện sẽ chạy qua giải thuật trích đặc trưng (**deep learning**) để có vector đặc trưng tương ứng.



Như vậy ta có vector đặc trưng của mỗi khuôn mặt cần nhận diện. Hình minh họa bên dưới:

Đây là ai?



Danh tính đã có



Nguyễn Hoàng Dương (B19DCCN153) 41:07

Ngân [1, 0, 0] **Minh** [0, 1, 0] **Nhung** [0, 0, 1]

Bảng việc so sánh vector đặc trưng của mỗi hình ảnh cần dự đoán **Đây là ai?** với **Danh tính đã có** bằng cách tính độ tương tự giữa hai vector dùng công thức ở mục phía trên:

Tính cosine similarity

Bài toán trên đã được số hóa thành:

Ngân = [1, 0, 0]

Minh = [0, 1, 0]

Nhung = [0, 0, 1]

Có một hình ảnh A sau khi được mã hóa là A = [0.9, 0.07, 0.03]

So trùng các vector A với các danh tính đã có Ngân, Minh, Nhung:

$$\cos(A, \text{Ngân}) = \frac{1 \times 0.9 + 0 \times 0.07 + 0 \times 0.03}{\sqrt{1^2 + 0^2 + 0^2} \times \sqrt{0.9^2 + 0.07^2 + 0.03^2}} = 0.9964$$

$$\cos(A, \text{Minh}) = \frac{0 \times 0.9 + 1 \times 0.07 + 0 \times 0.03}{\sqrt{0^2 + 1^2 + 0^2} \times \sqrt{0.9^2 + 0.07^2 + 0.03^2}} = 0.0775$$

$$\cos(A, \text{Nhung}) = \frac{0 \times 0.9 + 0 \times 0.07 + 1 \times 0.03}{\sqrt{0^2 + 0^2 + 1^2} \times \sqrt{0.9^2 + 0.07^2 + 0.03^2}} = 0.0332$$

Kết luận: A là **Ngân** vì có độ tương đồng lớn nhất.

Viết chương trình tính độ tương đồng cosin của hình ảnh mới với những người trong CSDL và in ra người có độ tương đồng lớn nhất trong hệ thống nhận diện danh tính khi biết vector đặc trưng của mỗi người và một bức ảnh mới A sau khi được mã hóa.

Input:

Dòng đầu tiên số nguyên N là tổng số dòng trong bộ test ($N > 3$).

Dòng thứ 2 là số tự nhiên n số chiều của vector đặc trưng cho khuôn mặt ($n < 1025$)

Dòng thứ 3 là gồm n số là chỉ số cho từng thành phần của vector đặc trưng cho một bức ảnh mới A

Nguyễn Hoàng Dương (B19DCCN153) 40:55

Input:

Dòng đầu tiên số nguyên N là tổng số dòng trong bộ test ($N > 3$).

Dòng thứ 2 là số tự nhiên n số chiều của vector đặc trưng cho khuôn mặt ($n < 1025$)

Dòng thứ 3 là gồm n số là chỉ số cho từng thành phần của vector đặc trưng cho một bức ảnh mới A

$N-3$ dòng còn lại bao gồm string S_n đầu tiên là tên người và sau đó là n số tương ứng với n chỉ số cho từng thành phần của vector đặc trưng cho người S_n đó

Output:

In ra tên người có vector tương đồng lớn nhất với A và in ra danh sách chỉ số độ tương đồng cosin của hình ảnh mới với những người trong CSDL làm tròn đến số thứ 4 sau dấu phẩy.

Trường hợp số nguyên $N < 4$ hoặc $n > 1024$ in ra **INVALID INPUT**

Ví dụ:

Input:

```
6
3
0.9 0.07 0.03
Ngan 1 0 0
Minh 0 1 0
Nhưng 0 0 1
```

Output:

```
Ngan [0.9964, 0.0775, 0.0332]
```

Giới hạn thời gian: 2s
Giới hạn bộ nhớ: 65536 Kb

Trình biên dịch

Python 3

Chọn tệp

Duyệt



Thực hành

Bài tập

Lịch sử

Hỗ trợ

Nguyễn Hoàng Dương (B19DCCN153) 40:33

TỪ ĐIỂN LỒNG NHAU

Tên và điểm thi môn Lập trình Python của mỗi sinh viên được lưu vào một từ điển, với key là tên sinh viên, value tương ứng là điểm thi của sinh viên. Lưu trữ chúng trong một từ điển lồng nhau. Với một lớp có N ($N \leq 70$) sinh viên, tìm và in ra tên của K ($K \leq 5$) sinh viên có điểm thi cao nhất. Nếu có nhiều sinh viên bằng điểm nhau, hãy sắp xếp tên theo chiều tăng dần của bảng chữ cái.

Input:

- Dòng đầu tiên là giá trị N và K , cách nhau bởi một khoảng trắng.
- N dòng tiếp theo là tên và điểm thi của sinh viên

Output:

- Tên của K sinh viên có điểm cao nhất, được viết trên một dòng duy nhất mỗi tên cách nhau bởi một khoảng trắng.
- Nếu giá trị N và K không thỏa mãn điều kiện, in ra thông báo: **INVALID INPUT**

Ví dụ:

Input	Output
10 3	
Hung 6	
Long 7	
Giang 8	
Linh 5	
Tuan 8	Ngan Hoa Khanh

Nguyễn Hoàng Dương (B19DCCN153) 40:28

- Tên của K sinh viên có điểm cao nhất, được viết trên một dòng duy nhất mỗi tên cách nhau bởi một khoảng trắng.
- Nếu giá trị N và K không thỏa mãn điều kiện, in ra thông báo: **INVALID INPUT**

Ví dụ:

Input	Output
10 3	
Hung 6	
Long 7	
Giang 8	
Linh 5	
Tuan 8	Ngan Hoa Khanh
Hoa 9	
Mai 5	
Ngoc 4	
Khanh 9	
Ngan 10	

Giới hạn thời gian: 2s

Giới hạn bộ nhớ: 65536 Kb

Trình biên dịch

Python 3

Chọn tệp

Duyệt

Nộp bài

Nguyễn Hoàng Dương (B19DCCN153) 33:11

MONTHLY SALARY

PHIẾU THU NHẬP THÁNG 10/2021

HỌ VÀ TÊN	Nguyễn Văn A	EMAIL	Nva@abc.com.vn
Chức danh	DEV	Bộ phận	IT
NGÀY CÔNG CHUẨN	22	CÁC KHOẢN KHẤU TRỪ (2)	000.000
Ngày công chính thức	22	BHXH (8%*LCB)	000.000
Ngày công thử việc	0	BHYT (1.5%*LCB)	00.000
THU NHẬP THỰC TẾ (1)	00.000.000	BHTN (1%*LCB)	00.000
Lương cơ bản chính thức (LCB)	0.000.000	Đoàn phí (1%*LCB)	00.000
Thưởng đánh giá tháng chính thức (LT)	00.000.000	Thuế THƯC	00.000
Lương cơ bản thử việc	0		
Thưởng đánh giá tháng thử việc	0		
Trợ cấp 1 (Không chịu thuế TNCN)			
Trợ cấp 2 (Không chịu thuế TNCN)			
THỰC LĨNH (3) = (1) - (2)	00.000.000		

Trên đây là Phiếu thu nhập của một người trong một tháng bao gồm:

- Thu nhập thực tế (1)
- Các khoản khấu trừ (2)
- Thực lĩnh (3)

Thu nhập thực tế (1) = Lương cơ bản (LCB) + Lương thưởng đánh giá tháng (LT)

Các khoản khấu trừ (2) = BHXH + BHYT + BHTN + ĐP + TNCN

Thực lĩnh (3) = Thu nhập thực tế (1) - Các khoản khấu trừ (2)

BHXH (Bảo hiểm xã hội) = 8%*LCB

Trần Quang Vinh (B19DCCN722)

DELTA SUBSTITUTION VARIANT

COVID-19 DELTA SUBSTITUTION VARIANT

Biến thể Delta của virus SARS-CoV-2 được phát hiện lần đầu tiên ở Ấn Độ vào tháng 10/2020. Biết rằng, cứ 11 ngày trong quần thể virus xảy ra một đột biến điểm (Nucleotide/ base).

Đây là thuật ngữ của di truyền học phân tử, trong tiếng nước ngoài gọi là **Point Mutation** (tiếng Anh) dùng để chỉ một loại đột biến gen chỉ xảy ra ở một "điểm" trên phân tử di truyền DNA. "Điểm" này tương ứng với một cặp Nucleotide duy nhất.



Hình 1: Bộ gen ban đầu của Virus

Có ba loại đột biến điểm – Point Mutation là:

- Đột biến thay thế - Substitution
- Đột biến xóa - Deletion
- Đột biến thêm - Insertion

Đột biến thay thế - Substitution:



Thay thế một số cặp Nucleotides AT thành GC hoặc thay đổi thứ tự AT thành TA và GC thành CG;

Nguyễn Hoàng Dương (B19DCCN153) 32:59

- Thu nhập thực tế (1)
- Các khoản khấu trừ (2)
- Thuế lĩnh (3)

Thu nhập thực tế (1) = Lương cơ bản (LCB) + Lương thưởng đánh giá tháng (LT)

Các khoản khấu trừ (2) = BHXH + BHYT + BHTN + ĐP + TNCN

Thuế lĩnh (3) = Thu nhập thực tế (1) - Các khoản khấu trừ (2)

BHXH (Bảo hiểm xã hội) = 8%*LCB

BHYT (Bảo hiểm y tế) = 1,5%*LCB

BHTN (Bảo hiểm thất nghiệp) = 1%*LCB

ĐP (Đón phụ) = 1%*LCB

Thuế TNCN (Thuế Thu nhập cá nhân) = Thu nhập chịu thuế (TNNT)*Thuế suất

Thu nhập chịu thuế được tính bằng:

TNNT = Thu nhập thực tế (1) - Các khoản giảm trừ - Tiền đóng bảo hiểm

Tiền đóng bảo hiểm (BH) = BHXH + BHYT + BHTN

Các khoản giảm trừ = Giảm trừ bản thân (11.000.000 VNĐ) + Người phụ thuộc (4.400.000VNĐ / 1 người)

Dựa vào điều kiện của Thu nhập chịu thuế (TNNT) như bảng dưới đây ta sẽ có cách tính Thuế TNCN:

Bậc	Thu nhập tính thuế /tháng	Thuế suất	Tính số thuế phải nộp	
			Cách 1	Cách 2
1	Đến 05 trđ	5%	0 trđ + 5% TNNT	5% TNNT
2	Trên 05 trđ đến 10 trđ	10%	0,25 trđ + 10% TNNT trên 5 trđ	10% TNNT - 0,25 trđ
3	Trên 10 trđ đến 18 trđ	15%	0,75 trđ + 15% TNNT trên 10 trđ	15% TNNT - 0,75 trđ
4	Trên 18 trđ đến 32 trđ	20%	1,95 trđ + 20% TNNT trên 18 trđ	20% TNNT - 1,65 trđ
5	Trên 32 trđ đến 52 trđ	25%	4,75 trđ + 25% TNNT trên 32 trđ	25% TNNT - 3,25 trđ
6	Trên 52 trđ đến 80 trđ	30%	9,75 trđ + 30% TNNT trên 52 trđ	30 % TNNT - 5,85 trđ
7	Trên 80 trđ	35%	18,15 trđ + 35% TNNT trên 80 trđ	35% TNNT - 9,85 trđ

Ví dụ: TNNT = 6.000.000VNĐ thì thuế TNCN = 250.000 + 10*(TNNT - 5.000.000) = 350.000VNĐ

Nguyễn Hoàng Dương (B19DCCN153) 32:50

4	Trên 10 trđ đến 32 trđ	20%	1,95 trđ + 20% TNNT trên 10 trđ	20% TNNT - 1,65 trđ
5	Trên 32 trđ đến 52 trđ	25%	4,75 trđ + 25% TNNT trên 32 trđ	25% TNNT - 3,25 trđ
6	Trên 52 trđ đến 80 trđ	30%	9,75 trđ + 30% TNNT trên 52 trđ	30 % TNNT - 5,85 trđ
7	Trên 80 trđ	35%	18,15 trđ + 35% TNNT trên 80 trđ	35% TNNT - 9,85 trđ

Ví dụ: TNNT = 6.000.000VNĐ thì thuế TNCN = 250.000 + 10*(TNNT - 5.000.000) = 350.000VNĐ

Viết chương trình in ra số tiền Thuế lĩnh của người lao động trong 1 tháng khi biết Thu nhập thực tế (1) và LCB của người đó hàng tháng và người này không có người phụ thuộc

Input:

Dòng đầu tiên số nguyên N (1 ≤ N ≤ 10) là số tập test.

N dòng tiếp theo, mỗi dòng gồm 2 số m, n lớn > 0. Trong đó m là Thu nhập thực tế (1) và n là LCB

Output:

In ra 1 số là tiền Thuế lĩnh hàng tháng (lấy đến phần nguyên)

Trường hợp số nguyên N vượt quá 10 hoặc nhỏ hơn 0 in ra **INVALID INPUT**

Bộ test nào có giá trị nhỏ hơn 1000 in ra dòng đó là **INVALID INPUT**

Ví dụ:

Input:	Output:
4	14773575
15700000 6300000	12559425
13300000 5700000	78097475
10000000 7700000	INVALID INPUT
1000 123	

Giới hạn thời gian: 2s

Giới hạn bộ nhớ: 65536 Kb



Nguyễn Hoàng Dương (B19DCCN153) 30:29

KHOẢNG CÁCH HAVERSINE

Khoảng cách Haversine là một đại lượng để xác định khoảng cách địa lý giữa hai địa điểm bất kỳ. Giả sử hai địa điểm P_1 và P_2 được quy đổi về kinh độ và vĩ độ tương ứng là $long_1, lat_1, long_2, lat_2$. Khoảng cách Haversine d được xác định theo công thức sau:

$$a = \sin^2\left(\frac{\Delta lat}{2}\right) + \cos(lat_1) \cdot \cos(lat_2) \cdot \sin^2\left(\frac{\Delta long}{2}\right)$$

$$c = 2 \cdot \arcsin(\sqrt{a})$$

$$d = R \cdot c$$

Trong đó:

$$\Delta lat = lat_2 - lat_1$$

$$\Delta long = long_2 - long_1$$

$R = 6731$ (km) (Bán kính Trái Đất)

Hãy viết hàm tính khoảng cách Haversine giữa hai địa điểm bất kỳ.

Input:

- Kinh độ và vĩ độ tương ứng của hai địa điểm, mỗi địa điểm được viết trên một dòng, mỗi giá trị cách nhau bởi khoảng trắng và là các số thực (float).

Output:

- Khoảng cách Haversine giữa hai địa điểm, kết quả được in ra với hai số sau dấu phẩy.

Ví dụ:

Input	Output
105.96 10.21	4686.47

Nguyễn Hoàng Dương (B19DCCN153) 30:22

Trong đó:

$$\Delta lat = lat_2 - lat_1$$

$$\Delta long = long_2 - long_1$$

$R = 6731$ (km) (Bán kính Trái Đất)

Hãy viết hàm tính khoảng cách Haversine giữa hai địa điểm bất kỳ.

Input:

- Kinh độ và vĩ độ tương ứng của hai địa điểm, mỗi địa điểm được viết trên một dòng, mỗi giá trị cách nhau bởi khoảng trắng và là các số thực (float).

Output:

- Khoảng cách Haversine giữa hai địa điểm, kết quả được in ra với hai số sau dấu phẩy.

Ví dụ:

Input	Output
105.96 10.21	4686.47
107.17 16.79	

Giới hạn thời gian: 2s

Giới hạn bộ nhớ: 65536 Kb

Trình biên dịch

Python 3

Chọn tệp

Duyệt

Nộp bài

HỆ SỐ JACCARD

Hệ số Jaccard là một đại lượng dùng để đo mức độ tương tự giữa hai chuỗi bất kỳ. Cho hai chuỗi A và B, hệ số Jaccard giữa A và B, được ký hiệu là $J(A, B)$ và được xác định theo công thức sau:

$$J(A, B) = \frac{|A \cap B|}{|A \cup B|} = \frac{|A \cap B|}{|A| + |B| - |A \cap B|}$$

Cho hai chuỗi A và B, Tính độ tương đồng Jaccard giữa hai chuỗi.

Lưu ý:

- Nếu A và B rỗng, thì $J(A, B) = 1$
- $J(A, B)$ nằm trong đoạn $[0, 1]$
- Lưu ý các ký tự trong chuỗi để tính tương đồng không phân biệt chữ hoa thường, không bao gồm các ký tự như khoảng trắng, dấu chấm (.), dấu phẩy (,) và dấu hỏi chấm (?).

Input:

Hai chuỗi A và B, mỗi chuỗi được viết trên một dòng.

Output:

Hệ số Jaccard giữa A và B. Kết quả được in ra với 2 chữ số sau dấu phẩy.

Ví dụ:

Input	Output
Hôm nay thi lập trình python	0.91
Lập trình Python.	
Hôm qua trời mưa	0.50
Hôm qua mưa, ngày mai có nắng không?	

Giới hạn thời gian: 2s

Giới hạn bộ nhớ: 65536 Kb

Trình biên dịch

Python 3

Chọn tệp

Duyệt

DANH SÁCH TUPLES

Cho một danh sách A[] gồm N tuples, mỗi tuple không quá 10 phần tử. Hãy kết hợp các tuples có phần tử đầu tiên giống nhau.

Input:

- Dòng đầu là số N ($N < 100$)
- N dòng tiếp theo là N tuples của mảng A, mỗi tuple được viết trên một dòng

Output:

- Mảng gồm các tuples sau khi đã kết hợp.

Ví dụ:

Input	Output
4	
5 6	
5 7	[(5, 6, 7, 8), (6, 5, 8, 7)]
5 8	
6 5 8 7	
3	
6 7	
6 10	[(6, 7, 10, 8)]
6 8	

Giới hạn thời gian: 2s

Giới hạn bộ nhớ: 65536 Kb

Trình biên dịch

Python 3

Chọn tệp

Duyệt