**Phương pháp**

Ý tưởng chính của phương pháp là tính toán sự tương tự giữa 2 câu dựa trên các mức độ trừu tượng khác nhau của các câu đầu vào.

Cho 1 tập các cặp câu đã được gán nhãn {<S1,1, S1,2, y1>, ..., <Sn,1, Sn,2, yn>}.

Trong đó:

* Si,1 và Si,2 là cặp câu thứ i.
* yi nhận giá trị 1 nếu hai câu có nghĩa tương tự, 0 nếu ngược lại.

Mỗi cặp câu được biểu diễn bằng một vectơ đặc trưng vi, giá trị của vectơ được trả về bởi độ đo sự tương tự - cái mà chỉ ra mức độ tương tự giữa hai câu Si,1 và Si,2 ở các mức độ trừu tượng khác nhau...

9 độ đo chuỗi tương tự bao gồm khoảng cách Levenshtein, khoảng cách Jaro-Winkler, khoảng cách Manhattan, khoảng cách Euclide, độ tương tự cosin, khoảng cách n-gram (n = 3), hệ số phù hợp, hệ số Dice và hệ số Jaccard.

**7 mức độ trừu tượng**

Mỗi cặp câu đầu vào, hình thành 7 cặp chuỗi mới:

1. 2 chuỗi bao gồm các âm tiết của S1 và S2 tương ứng với vị trí ban đầu của mỗi “token”.
2. Như trường hợp trước nhưng giờ các “token” được thay bằng các từ của chúng
3. Như trường hợp trước nhưng giờ các từ được thay bằng nhãn từ loại của chúng
4. Hai chuỗi bao gồm các nhãn từ loại danh từ, động từ và tính từ của S1 và S2, với thứ tự đúng như thứ tự ban đầu của các danh từ, động từ và tính từ trong câu.

Hai chuỗi bao gồm các danh từ, động từ, tính từ của S1, S2 các từ này dk xác định nhờ vào nhãn từ loại của chúng.

1. Như trường hợp trước nhưng chỉ giữ lại các danh từ
2. Như trường hợp 4 nhưng chỉ giữ lại các động từ
3. Như trường hợp 4 nhưng chỉ giữ lại các tính từ

* Tổng cộng, 9 độ đo sự tương tự kết hợp với 7 cặp chuỗi cho ta 63 giá trị.

**Độ đo sự tương tự giữa 2 chuỗi S1 và S2**

1. Khoảng cách Jaro-Winkler

* Khoảng cách Jaro (dj)



Trong đó:

* m: số ký tự *phù hợp* (ký tự xuất hiện ở cả 2 chuỗi S1 và S2 và khoảng cách giữa vị trí của 2 ký tự ở này không được vượt quá



* t: ½ số ký tự *chuyển vị* (số lượng ký tự chuyển vị là số lượng ký tự *phù hợp* mà thứ tự của các ký tự bị thay đổi (khác nhau) trong hai chuỗi.
* Khoảng cách Jaro-Winkler (dw)

dw = dj + (lp(1 - dj))

Trong đó:

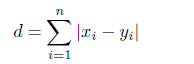
* dj là khoảng cách Jaro giữa hai chuỗi S1 và S2
* l là số lượng ký tự của chuỗi cùng xuất hiện ở đầu 2 chuỗi (tiền tố) nhưng giá trị lớn nhất chỉ là 4.
* p = 0.1

1. Khoảng cách Levenshtein

* Là số bước ít nhất biến chuỗi S1 thành S2 thông qua 3 phép biến đổi là:
* xóa 1 ký tự
* thêm 1 ký tự
* thay ký tự này bằng ký tự khác
* Thuật toán: <https://vi.wikipedia.org/wiki/Kho%E1%BA%A3ng_c%C3%A1ch_Levenshtein>

1. Khoảng cách Manhattan

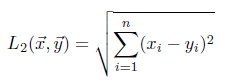
Công thức tính khoảng cách giữa 2 điểm X = (X1, X2, ...) và điểm Y = (Y1, Y2, ...) là:



Trong đó:

* n: số lượng từ khác nhau trong cả 2 câu (chuỗi) S1 và S2.
* *xi* và *yi* : số lần xuất hiện của mỗi từ tương ứng

1. Khoảng cách Euclide



1. Độ tương tự cosin



Trong đó, các vectơ  như trên nhưng các giá trị *xi* và *yi* là giá trị nhị phân (tùy thuộc vào từ tương ứng có xuất hiện hay không xuất hiện ở câu thứ nhất và câu thứ hai)

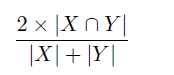
1. Khoảng cách n-gram

Giống như khoảng cách Manhattan nhưng thay vì sử dụng từ chúng ta sử dụng tất cả các ký tự n-gram trong 2 chuỗi. (n = 3)

1. Hệ số phù hợp

Đây là một hệ số phù hợp (matching) đơn giản được tính bằng cách đếm xem có bao nhiêu từ chung ở 2 câu.

1. Hệ số Dice



1. Hệ số Jaccard

