1. ****TF-IDF****

****TF****: Term Frequency(Tần suất xuất hiện của từ) là số lần từ xuất hiện trong văn bản. Vì các văn bản có thể có độ dài ngắn khác nhau nên một số từ có thể xuất hiện nhiều lần trong một văn bản dài hơn là một văn bản ngắn. Như vậy, term frequency thường được chia cho độ dài văn bản( tổng số từ trong một văn bản).

tf(t,d) = f(t,d)/(max{f(w,d) : w thuộc d}

Trong đó:

+ tf(t,d): tần suất xuất hiện của từ t trong văn bản d

+ f(t,d): Số lần xuất hiện của từ t trong văn bản d

+ max({f(w,d):w thuộc d}): số lần xuất hiện của từ có số lần xuất hiện nhiều nhất trong văn bản d

****IDF****: Inverse Document Frequency(Nghịch đảo tần suất của văn bản), giúp đánh giá tầm quan trọng của một từ . Khi tính toán TF , tất cả các từ được coi như có độ quan trọng bằng nhau. Nhưng  một số từ như “is”, “of” và “that” thường xuất hiện rất nhiều lần nhưng độ quan trọng là không cao. Như thế chúng ta cần giảm độ quan trọng của những từ này xuống.

Idf(t,D) = log(|D| / |{d thuộc D: t thuộc d}|)

Trong đó :

+ idf(t,D): Giá trị idf của từ t trong tập văn bản

+ |D| : tổng số văn bản trong tập D

+ |{d thuộc D: t thuộc d}| : thể hiện số văn bản trong tập D có chứa từ t

=> tf-idf(t,d,D) = tf(t,d)\*idf(t,D) là Trọng số được sử dụng để đánh giá tầm quan trọng của một từ trong một văn bản. Giá trị cao thể hiện độ quan trọng cao.

1. **K-means++**

- Giúp tìm khởi tạo K trung tâm ban đầu của cụm tốt hơn là random.

Các bước thực hiện:

B1: Chọn ngẫu nhiên trọng tâm đầu tiên từ các điểm dữ liệu.

B2: Đối với mỗi điểm dữ liệu tính khoảng cách nhỏ nhất từ nó đến các trọng tâm được chọn trước đó.

B3: Chọn trọng tâm tiếp theo từ các điểm dữ liệu sao cho xác suất chọn một điểm là trọng tâm tỷ lệ thuận với khoảng cách của nó so với trọng tâm được chọn trước đó. (tức là điểm có khoảng cách lớn nhất từ tâm gần nhất rất có thể được chọn tiếp theo là một trọng tâm)

B4: Lặp lại các bước 2 và 3 cho đến khi lấy đủ K trọng tâm.

1. **Nhận xét dữ liệu**

- Vì là bài phân cụm nên em phân cụm trực tiếp trên tập dữ liệu test

- Mỗi điểm dữ liệu có vector ví dụ một vector có dạng a = (0, 0, 0, 4, 0, 7, 8, 0, 0, 1) sẽ được biểu diễn thành a = 3:4 5:7 6:8 9:1. Vector có 10290 chiều tương đương với 10290 từ trong văn bản và giá trị là **TF-IDF của từ đó.**

**- Có 7532 điểm dữ liệu**

**- Bài sẽ phân ra làm 20 cụm tương ứng với 20 chủ đề của văn bản**

- Nhận xét dữ liệu: mỗi điểm dữ liệu là vector x, khảo sát có ||x|| = 1

=> ta sử dụng khoảng cách cosin để thực hiện tính khoảng cách giữa 2 điểm giữ liệu (khoảng cách cosin làm việc tốt với TF-IDF).Xét vector x,y

Có cos(x,y) = (x\*y)/căn(||x||\*||y||)

mà 2 điểm dữ liệu càng giống nhau

<=> góc nhỏ

<=> (x\*y)/căn(||x||\*||y||) nhỏ

<=> x\*y nhỏ do ||x|| = ||y|| = 1 với x,y là điểm dữ liệu trong file test

- Vì các vector đều có độ dài bằng nhau và bằng 1. Nên tính trung bình cụm có thể tính bằng trung bình.

**Thực hiện phân cụm trên tập data test:**

B1: Xác định 20 điểm trọng tâm ban đầu sử dụng k-mean++ để tìm

B2: xác định các điểm dữ liệu thuộc cụm nào bằng cách tính khoảng cách từ điểm dữ liệu đó đến các trọng tâm, gần trọng tâm nào nhất thì thuộc vào cụm đó

B3: Sau đó cập nhập lại trọng tâm của cụm bằng trung bình các điểm ở trong cụm mà mình của xếp

B4: thực hiện lại bước 2 và bước 3 đến khi nào trọng tâm cụm không thay đổi

1. **Nhận xét kết quả**

- Do ta đã biết số cụm là 20 nên có thể sử dụng Purity (độ tinh khiết) để đánh giá cụm dựa vào nhãn được cung cấp

-Cách Purity nhận xét cụm dựa trên nhãn:

+Sau khi phân cụm xong ta lấy nhãn của dữ liệu, và tìm nhãn xuất hiện nhiều nhất trong mỗi cụm. Ta gắn cụm đó với nhãn xuất hiện nhiều nhất

+ Sau đó ta tính xác suất M/N (với M là số lược điểm dữ liệu ở đúng cụm, N là tổng số điểm dữ liệu)

- Độ tinh khiết đạt được : 40.65321295804567 (cái này chỉ mang tính chất tham khảo vì khi cài đặt tính có thể sai sót và có nhiều tiêu chuẩn để đánh giá cụm khác nhau)

1. **Ý tưởng cải tiến**

- Có chiến thuật chọn k điểm ban đầu tốt hơn

- Theo đề bài thì các bài báo kia cũng có thể phân thành 6 cụm. Ta có thể dùng k-mean phân thành 6 cụm rồi mỗi cụm đấy tiếp dùng dùng k-mean phân cụm tiếp để được 20 cụm