

# 6. テキストデータベース

(マルチメディアデータベース序論, 全6回)

<https://www.kkaneko.jp/de/multimediatdb/index.html>

金子邦彦



# テキスト検索の技術



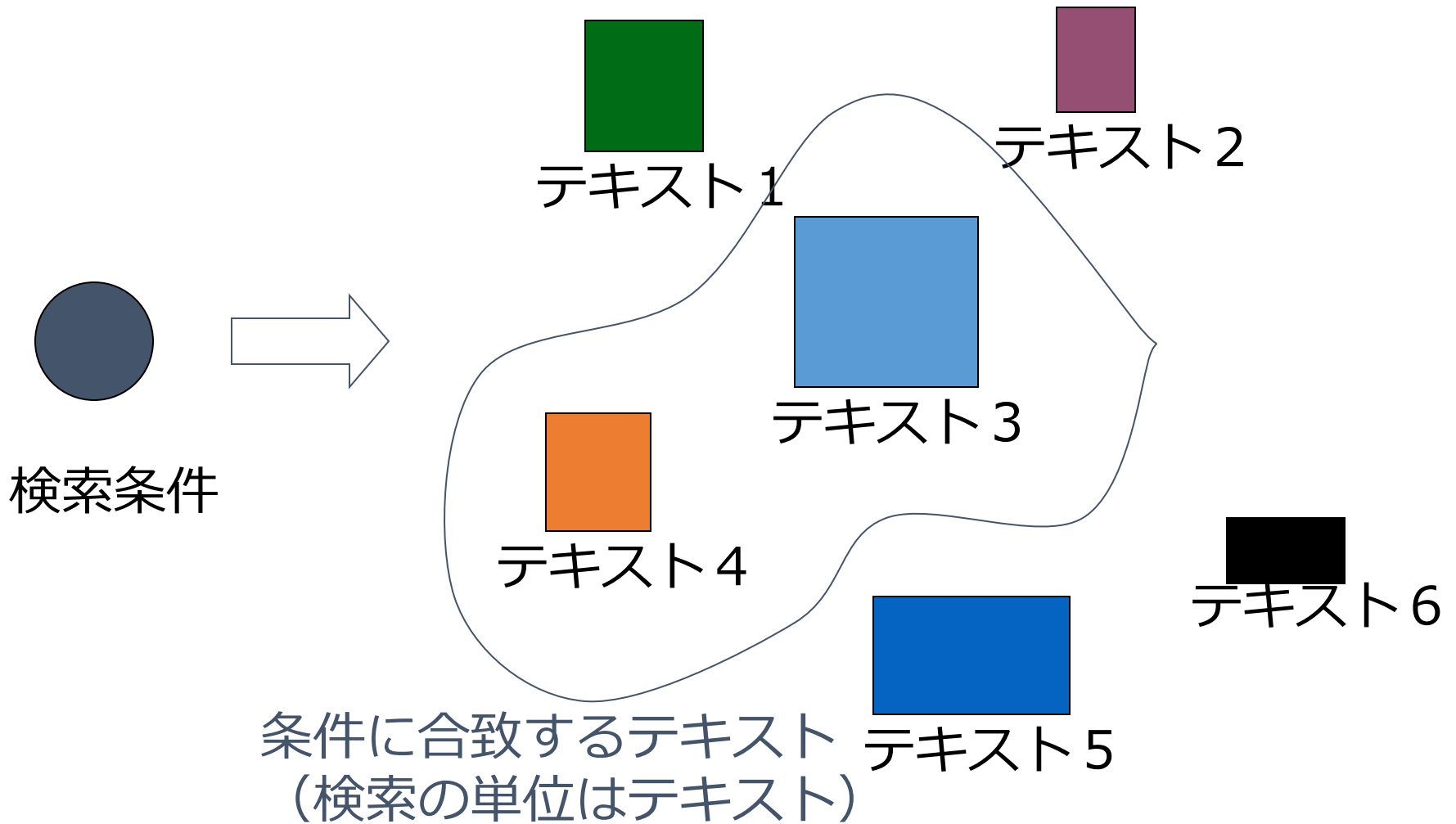
- テキストデータのベクトル表現
- 検索

# テキスト検索を行う局面



- 図書館で本を探す
- 特許出願で，関連の特許を探す
- 論文執筆で，関連研究を探す
- 新聞等から株価情報を抜き出す
- WWWを使って，興味のある情報を探す
- 判例を探す

# テキスト検索



# テキストのベクトル表現例



	テキスト 1	テキスト 2	テキスト 3	テキスト 4	テキスト 5
term 1	あり		あり	あり	
term 2		あり			
term 3	あり		あり		
term 4	あり	あり		あり	あり
term 5	あり	あり		あり	
term 6	あり				
term 7		あり			

これで1ベクトル

# テキストのベクトル表現



- term の有無や登場回数を使って、ベクトル表現
  - 非常に長いベクトルで表現（理由： キーワードの数が多い）
- 「検索条件」も、キーワードによるベクトル表現
  - 類似検索を、ベクトルマッチングで行う
  - 検索時には、「term ごとに重要度を変えたい」こともある

# content word とは



- content word
  - 検索に使う単語
  - テキスト中の有無／登場回数を使って，検索を行う
- non-content word
  - 検索に使わない単語
  - of, a, 「の」, 「が」 など

# テキストのベクトル表現例



- テキスト

[f1 ... fi ... fn]

n : term の総数

fi : i番目の term の有無 / 登場回数

- 問い合わせ

[d1, ..., di, ..., dn]

di : i番目の term の重要度



# document frequency



- term (Xとする) について, Xが登場する文章の数を document frequency という
  - document frequency は term ごとに定まる値

# document frequency



- document frequency が低い
  - あまり多くの文章に登場する
  - 「文章を区別するのに役に立つ term だ」と考える
- document frequency が高い
  - たくさんの文章に登場する

# inverse document frequency (idf)



- $\log (m/d)$

m: document の総数

d: term の document frequency

d=m ならば  $\log(m/d) = 0$

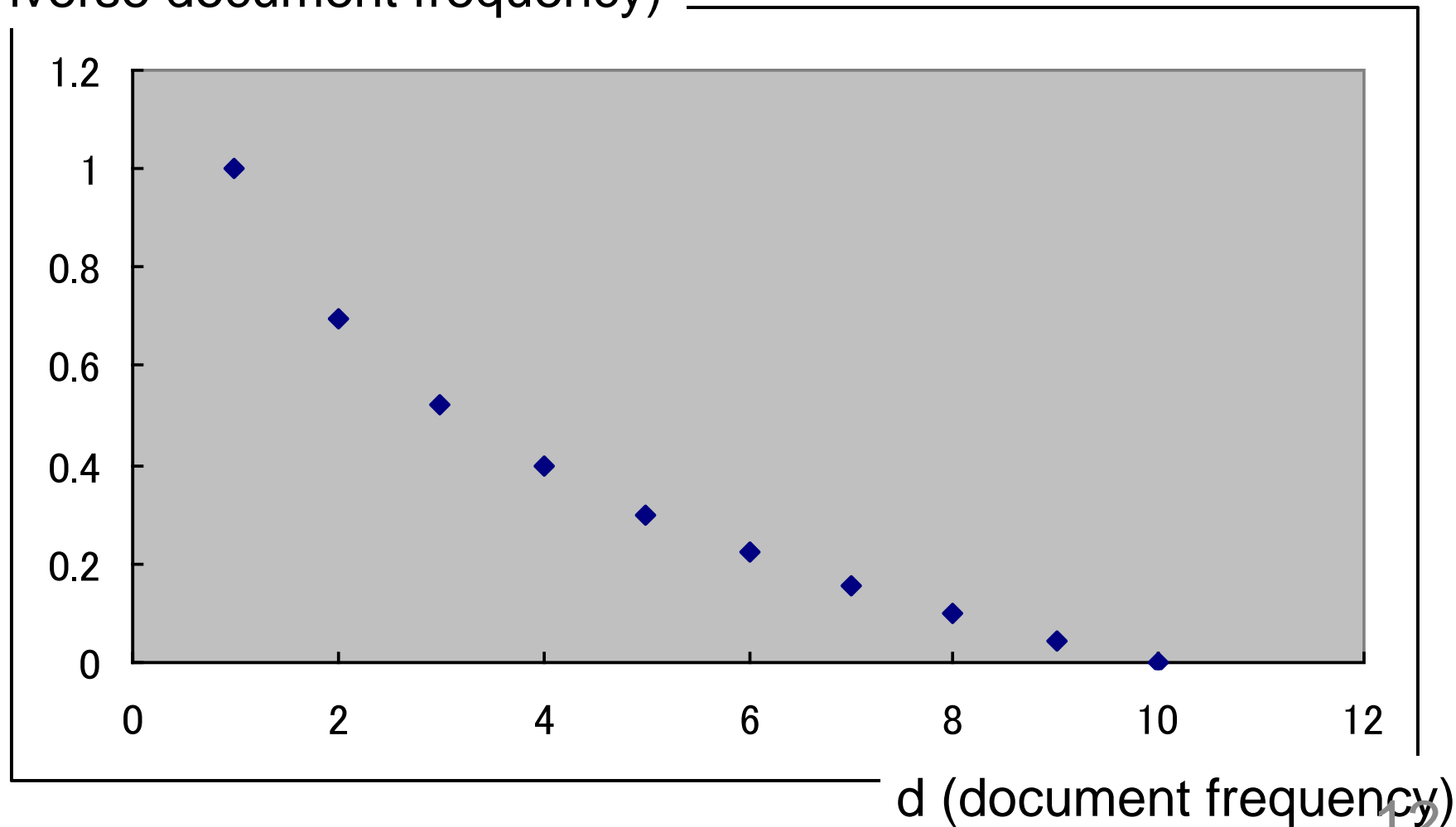
d=1 ならば  $\log(m/d) = \log m$

# log (m/d) のグラフ



m=10 のとき Datase Lab.

log (m/d)  
(inverse document frequency)



# term occurrence frequency(tf)

- ある term が, 文章中に登場する回数のこと
- term occurrence frequency が高いと
  - その term が何度も使われている
  - 筆者は, 意図して何度も使っているはず
  - その文章において, その term は, 「重要度が高い」と考える

$$f \cdot \log (m/d)$$

m: document の総数

d: term の document frequency

f: term の term occurrence frequency

単語ごと, 文章ごとに定まる値

# テキストのベクトル表現例

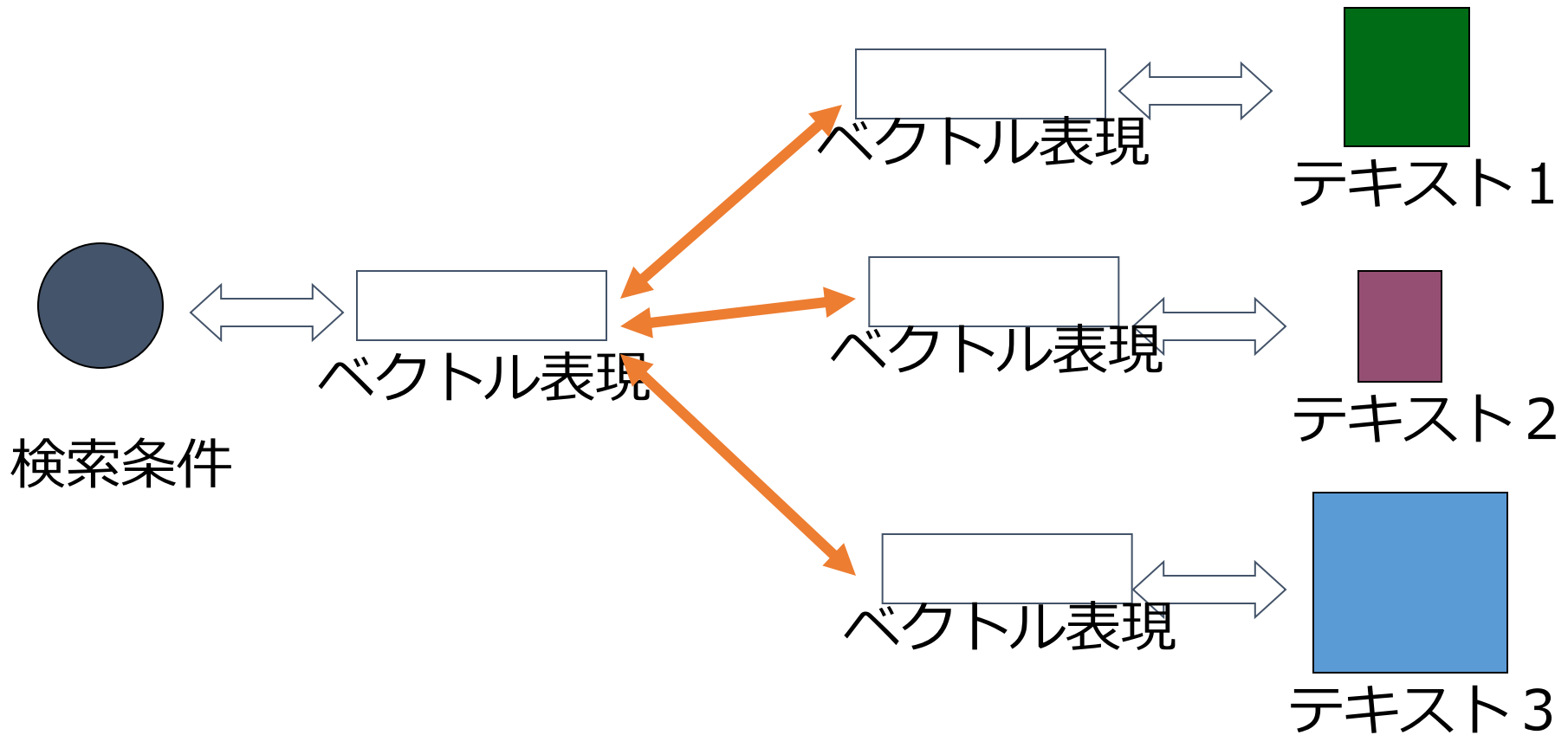


テキスト [x1 ... xi ... xn]

n : term の総数

xi : i番目の term の「tf/idf」値

# Retrieval



各々、ベクトルマッチングを行い、  
ベクトル空間中での「距離」に近いもの同士を  
類似度が高いとみなす (→解とする)



# ベクトルの距離



- dot product による距離

$$x_1y_1 + x_2y_2 + \cdot \cdot \cdot + x_ny_n$$

●  $(x_1, x_2, \dots, x_n)$

●  $(y_1, y_2, \dots, y_n)$

# dot product を使用しない理由



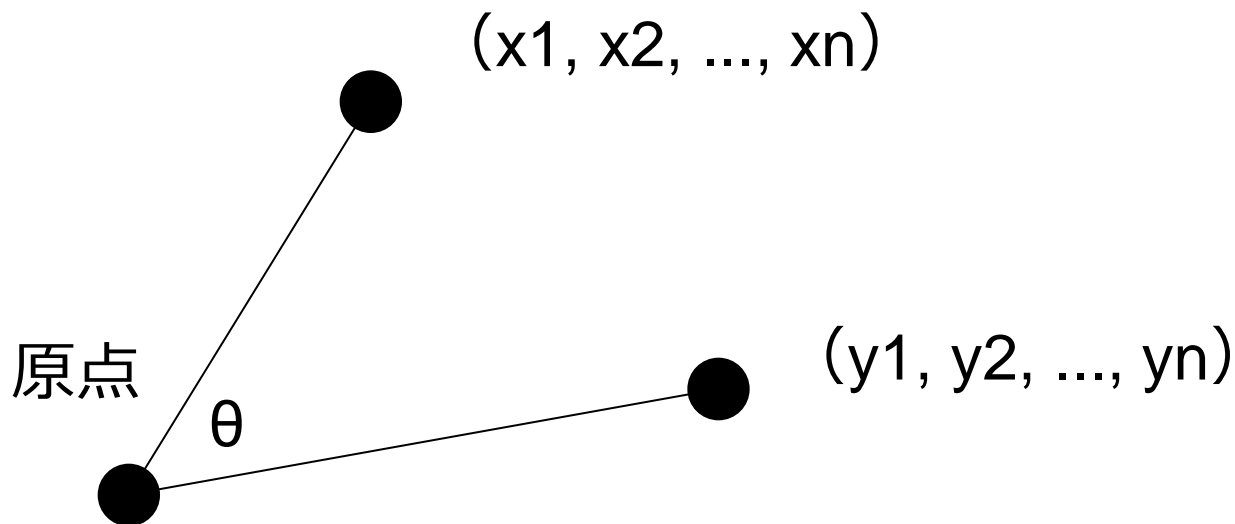
- 各  $x_i$  値は, tf/idf 値
    - 長い文章と短い文章では, 長い文章の方が tf/idf 値が大きくなって, マッチしやすくなる
- dot product による距離に代わる何かが必要

# Cosine距離



- $\text{Cosine}\theta$  のこと (2つのベクトルのなす角:  $\theta$ )
- $\text{Cosine}(X, Y) = \text{Cosine}(cX, Y) = \text{Cosine}(X, cY)$

$$\text{Cosine}(X, Y) = \frac{X \cdot Y}{\sqrt{(X \cdot X) \cdot (Y \cdot Y)}}$$



# テキスト検索における課題

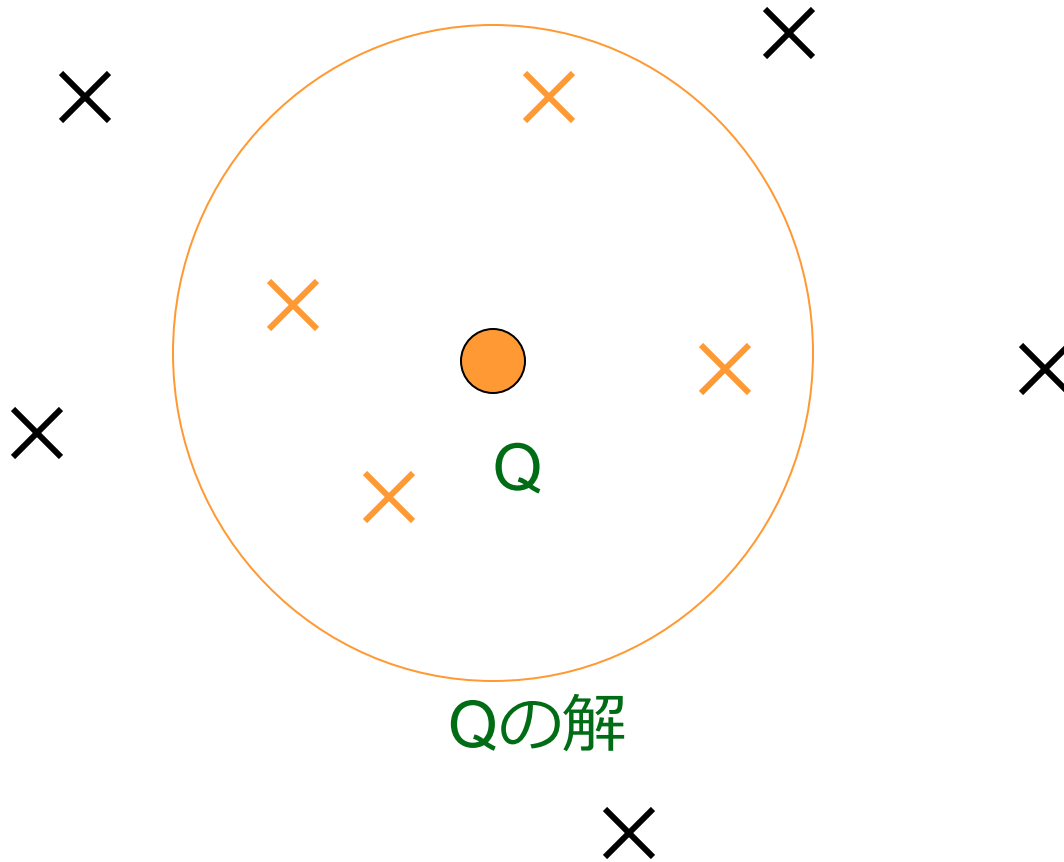


- Relevance Feedback
- インデックス
- tf/idf 以外のベクトル表現法

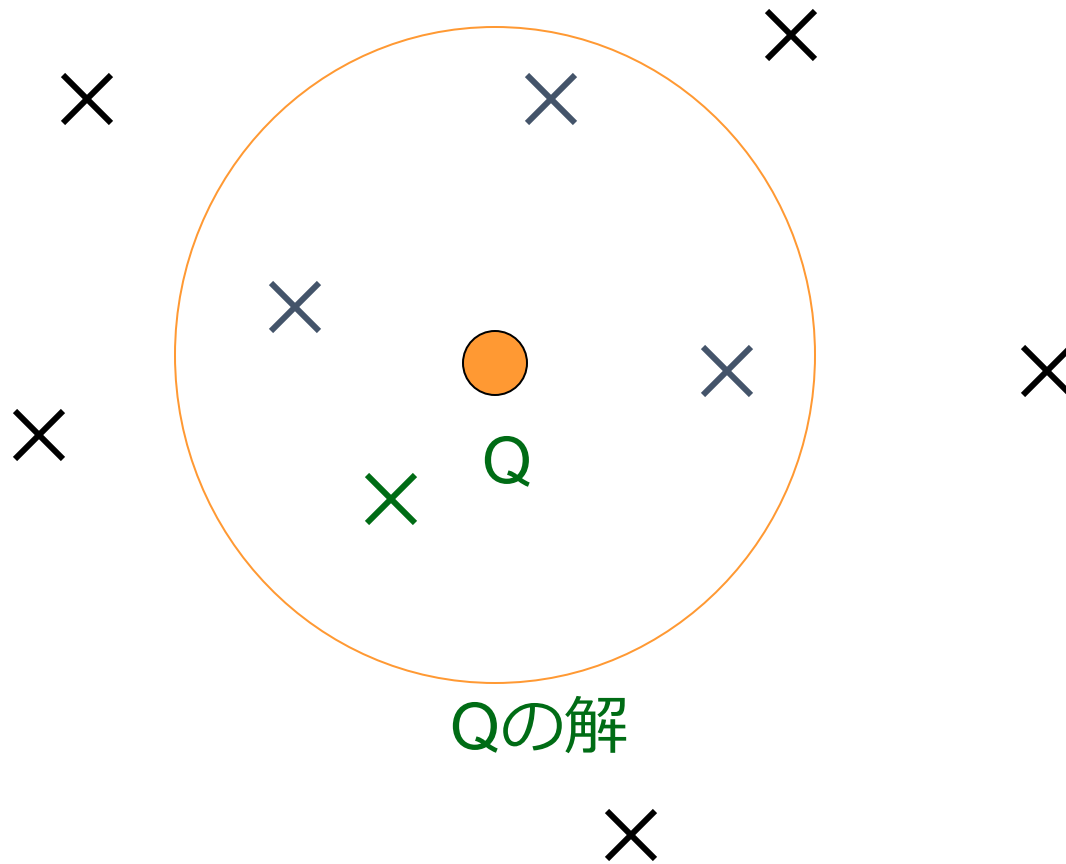
など



# Relevance Feedback(1/3)

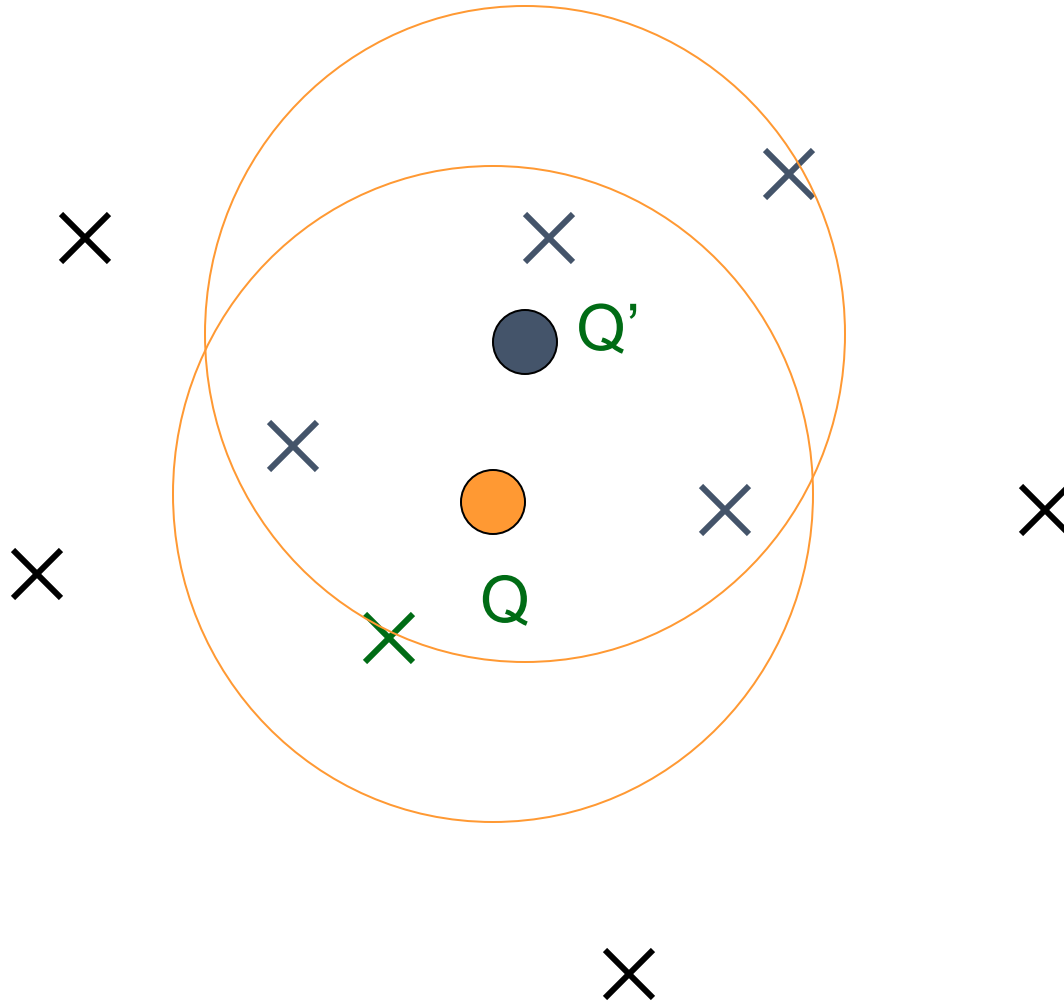


# Relevance Feedback(2/3)



ユーザは、どれが正しくて、どれが正しくないか分かる

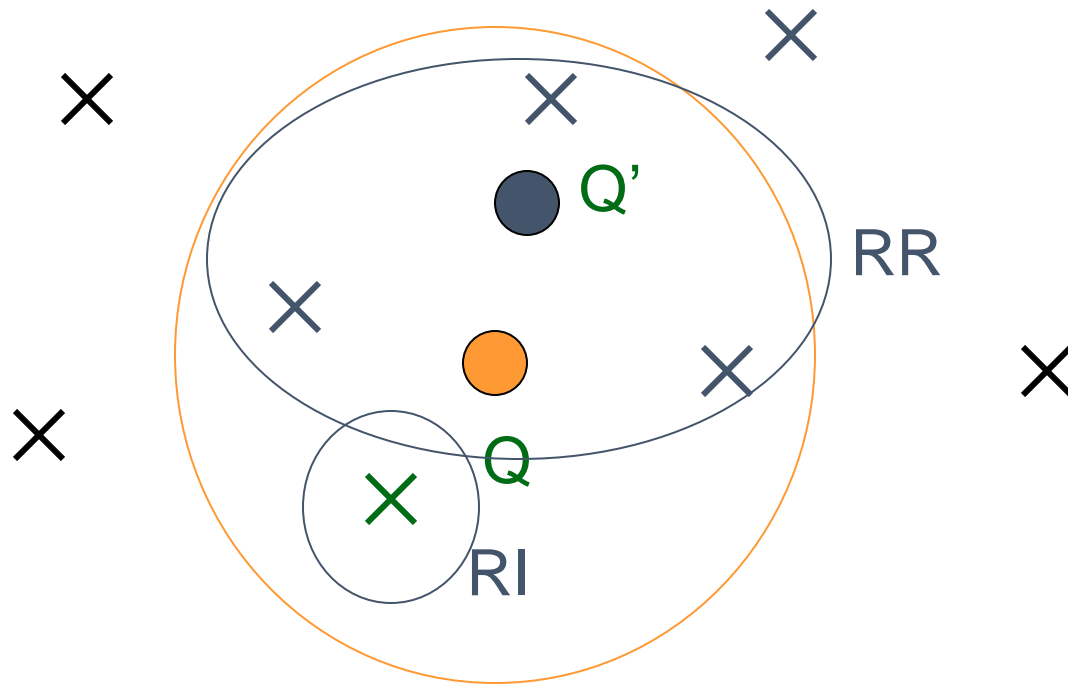
# Relevance Feedback(3/3)



システムは、新しい  $Q'$  を自動的に求め、再度問い合わせを実行

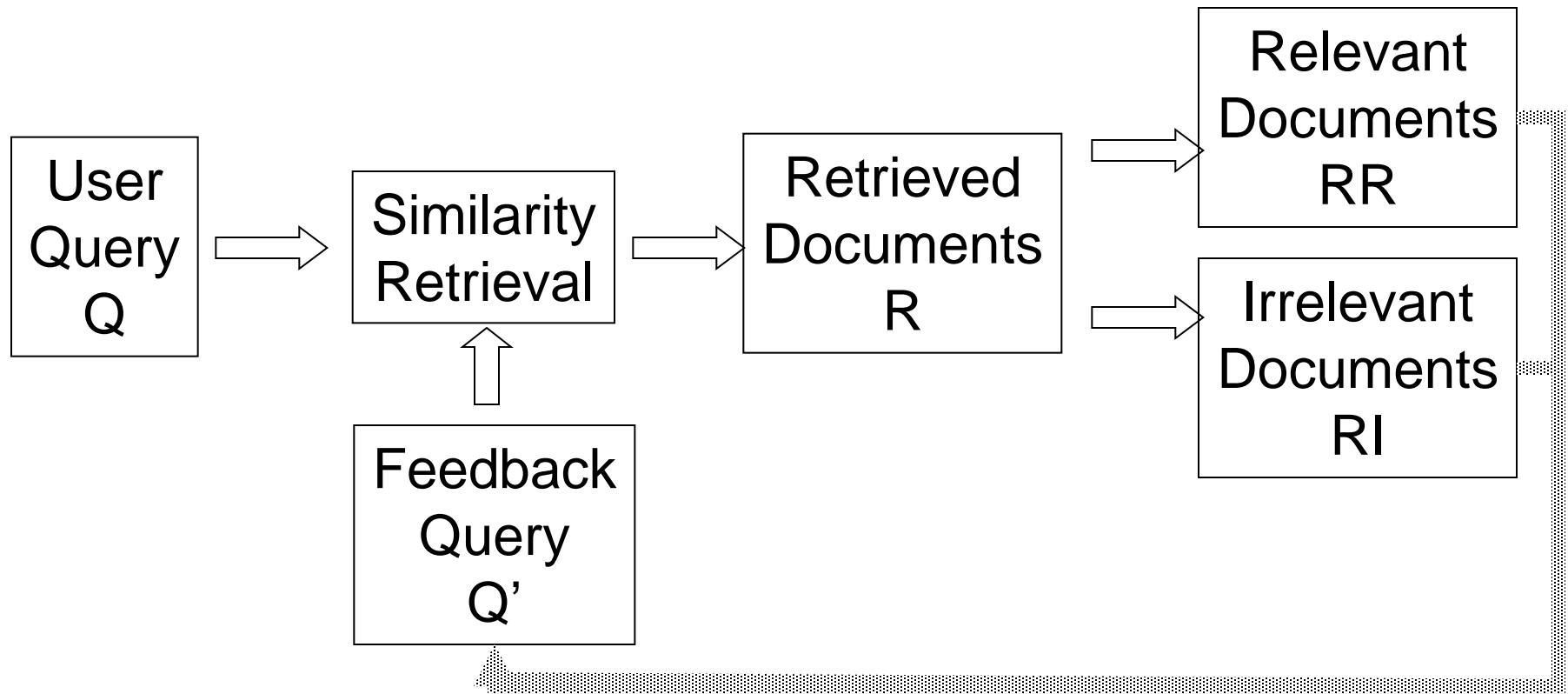


# Relevance Feedback



$$Q' = Q + C1 \cdot f(RR) - C2 \cdot f(RI)$$

# Relevance Feedback

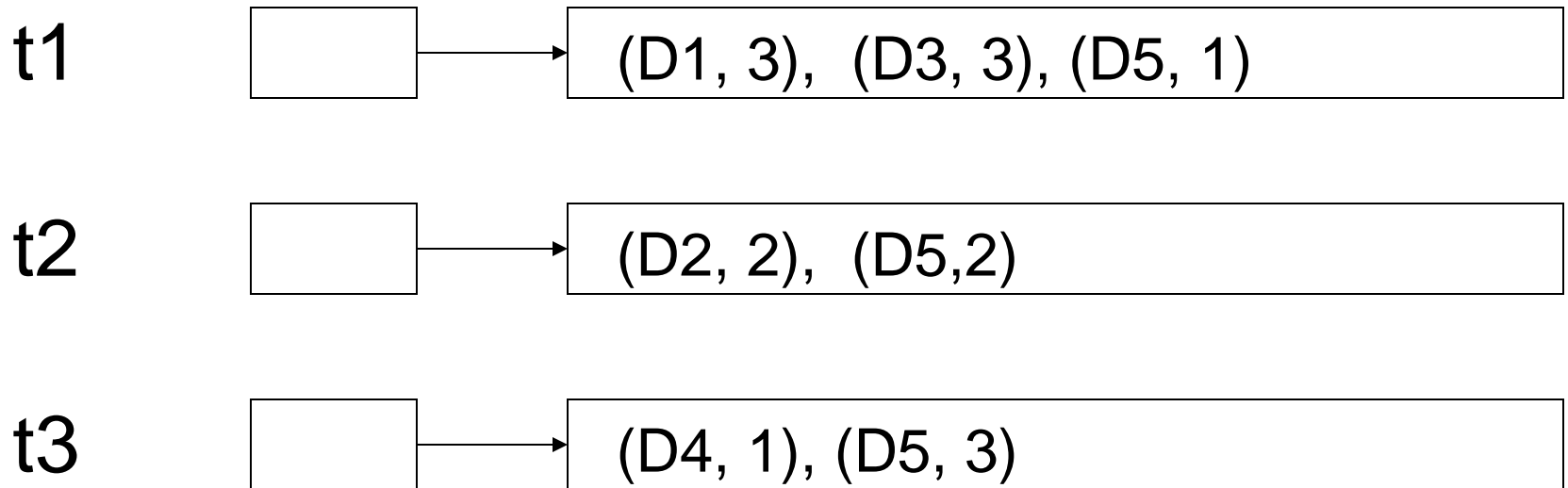


# インデックス



- inverted file
- signature file ← ハッシュを利用
- Clustering

# inverted file



term t3 は, D4, D5 へのみ登場し,  
それぞれのtf/idf 値は 1, 3

# inverted file

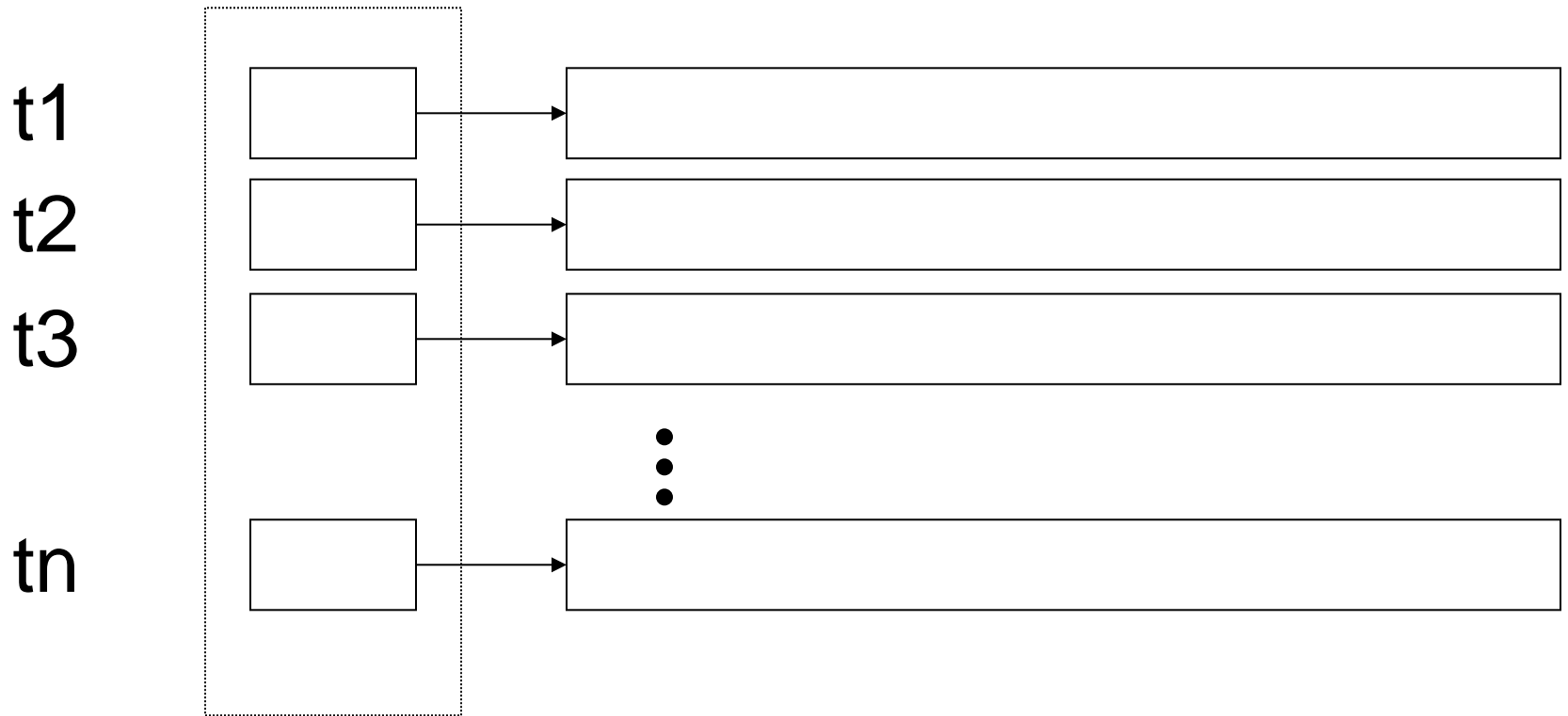


Q( 0, 2, 1 )に対して



「D2, D4, D5 のみ処理の対象とすべき」  
ことが分かる

# inverted file



この部分は普通 B+-tree

# ベクトル表現での課題



- 単語は違うが（ほぼ）同じ意味
  - 「おいしい」, 「美味しい」
  - 「不思議」, 「謎」
- 2単語で無く, 1単語とみなすべき
  - 「オペレーティング」, 「システム」
  - → 「オペレーティングシステム」

# ベクトル表現の限界



- 文章の意味には立ち入らない
  - 人が魚を食べた
  - 魚が人を食べた
- 登場する term は同じだが, 意味は違う