



# 情報アクセス論 第5回 「検索モデル」

情報理工学部  
前田 亮



## 第4回小テストの解説(1/4)

	$d_1$	$d_2$	$d_3$
$t_1$	1	0	0
$t_2$	0	1	2
$t_3$	1	1	0

上の表のように、各文書における索引語とその出現頻度が与えられているとする。このとき、文書  $d_2$  における索引語  $t_2$  の TF 値、索引語  $t_2$  の IDF 値、および文書  $d_2$  における索引語  $t_2$  の TF-IDF 値をそれぞれ求めよ。TF および IDF の計算には教科書の式 (4.2) (第4回配布資料の11枚目の式) および (4.4) (第4回配布資料の13枚目の式) を用いること。また、log の値は  $\log(3/2)=0.6$  を用いること。



## 第4回小テストの解説(2/4)

問1:

$$TF(t_2, d_2) = [A]$$

上式の[A]に入る値を答えよ.

$$\begin{aligned} TF(t, d) &= \frac{tf(t, d)}{\sum_{s \in d} tf(s, d)} && \cdots \text{教科書の式(4.2)(p.33)} \\ &= \frac{tf(t_2, d_2)}{tf(t_1, d_2) + tf(t_2, d_2) + tf(t_3, d_2)} \\ &= \frac{1}{0 + 1 + 1} \\ &= \frac{1}{2} \end{aligned}$$

	$d_1$	$d_2$	$d_3$
$t_1$	1	0	0
$t_2$	0	1	2
$t_3$	1	1	0



## 第4回小テストの解説(3/4)

問2:

$$IDF(t_2) = \log[B]$$

上式の[B]に入る値を答えよ.

	$d_1$	$d_2$	$d_3$
$t_1$	1	0	0
$t_2$	0	1	2
$t_3$	1	1	0

$$\begin{aligned} IDF(t) &= \log \frac{N}{df(t)} && \cdots \text{教科書の式(4.4)(p.34)} \\ &= \log \frac{3}{df(t_2)} \\ &= \log \frac{3}{2} \end{aligned}$$



## 第4回小テストの解説(4/4)

問3:

$$TF \cdot IDF(t_2, d_2) = [C]$$

上式の[C]に入る値を答えよ.

	$d_1$	$d_2$	$d_3$
$t_1$	1	0	0
$t_2$	0	1	2
$t_3$	1	1	0

$$\begin{aligned} TF \cdot IDF(t_2, d_2) &= TF(t_2, d_2) \cdot IDF(t_2) \\ &= \frac{1}{2} \cdot 0.6 \\ &= 0.3 \end{aligned}$$



## 検索モデルとは

- 文書(索引語の集合)と問合せとのマッチングやランキングを行う手法
  - ブーリアンモデル
  - ベクトル空間モデル
  - 確率モデル
  - 他にも様々な検索モデルが提案されているが、代表的なものは上の3つ



# ブーリアンモデルとは

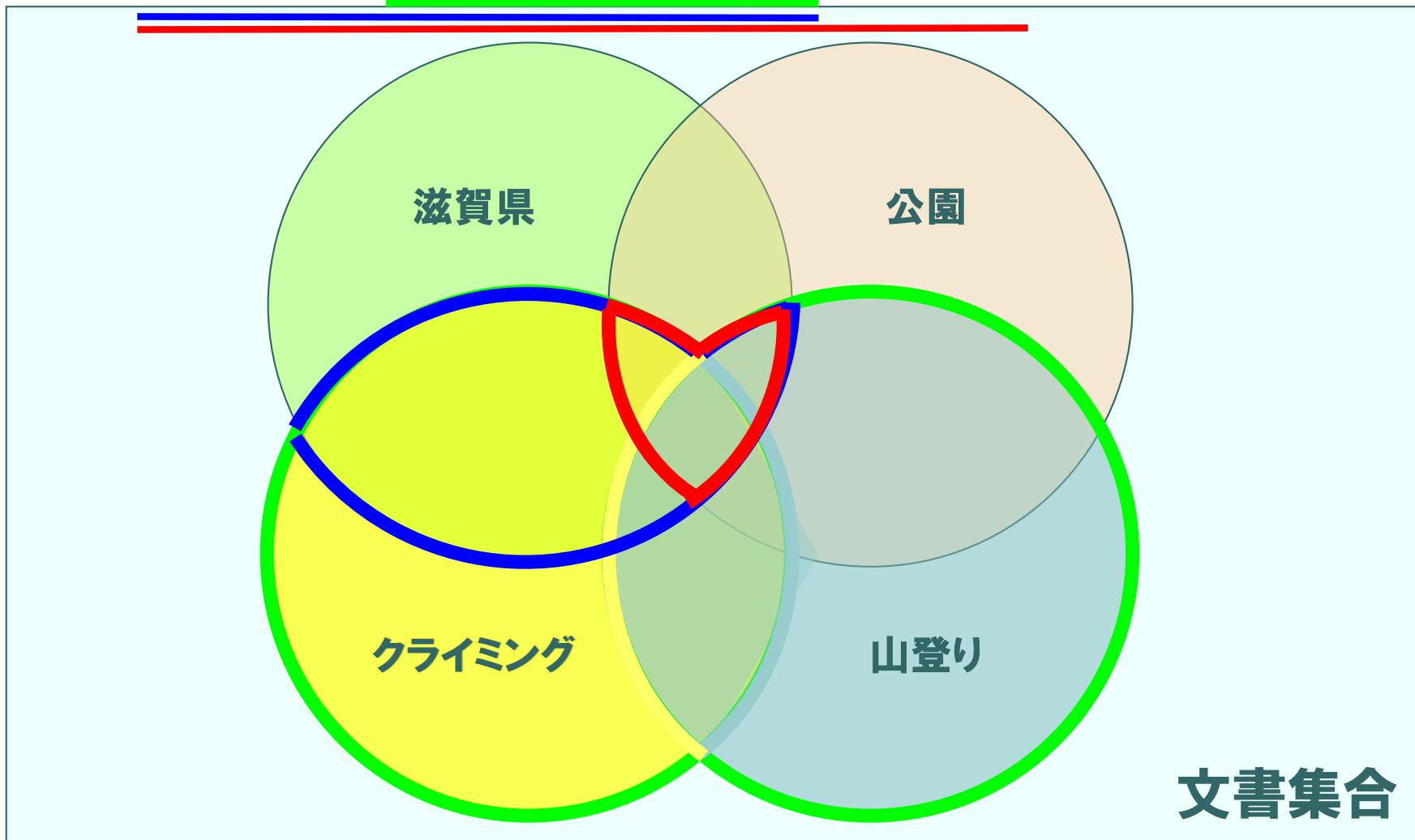
- 問合せを、ブール代数(演算子**AND**, **OR**, **NOT**と、値**0**, **1**のみをとる変数からなる)に基づく論理式で表現
  - 論理型モデルとも呼ばれる
- 例:「滋賀県でクライミングか山登りができる公園」を知りたい
- 問合せ:滋賀県 **AND** (クライミング **OR** 山登り) **AND** 公園





# ブーリアンモデルによる検索

問合せ: 滋賀県 AND (クライミング OR 山登り) AND 公園



# ブーリアンモデルを用いた検索の例

問合せ:滋賀県 AND (クライミング OR 山登り) AND 公園

〈文書 1〉

滋賀県にオープンした新しいクライミング公園! 開園記念イベント…

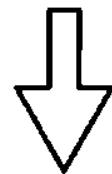


滋賀県  
クライミング  
山登り  
公園

返却される

〈文書 2〉

滋賀県に昔からあり、山登りができる〇〇公園です。

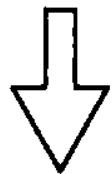


滋賀県  
山登り  
公園

返却される

〈文書 3〉

滋賀県でお薦めのクライミング、山登りができる公園はこちらです。



滋賀県  
クライミング  
山登り  
公園

返却される

〈文書 4〉

クライミングも良いけれど、山登りもいいよね。そういうとき…



クライミング  
山登り  
公園

返却されない



## ブーリアンモデルの利点

- 検索語同士の関係を明示的に記述でき、複雑な検索要求にも対応できる
  - 求める情報を検索語で表現し、その関係をAND, OR, NOTにより表現
  - 問合せを直感的に作成しやすい



## ブーリアンモデルの欠点

- 検索語を「含む」か「含まない」の2択しかない
  - 検索語を「部分的に含む」や、検索語に「類似した語を含む」などの要求に対応できない
- 文書中の検索語の頻度を考慮しない
  - 先の例では「公園」を探しているので、「公園」について多数言及している文書がほしい
- 問合せとの近さに基づく検索結果のランキングができない

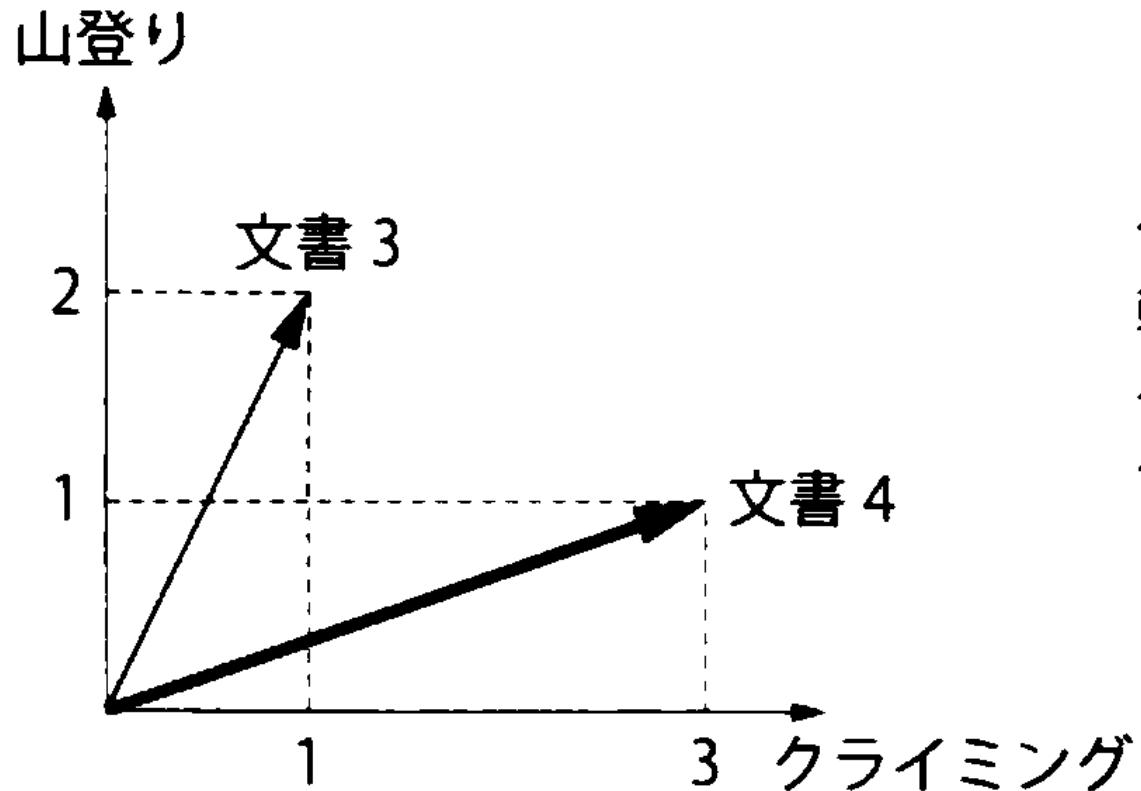


# ベクトル空間モデルとは

- 文書を単語の塊とみなし、ベクトルとして表現
  - 文書ベクトル
- 問合せも同様に、ベクトルとして表現
  - 問合せベクトル
- 上記二つのベクトルの向きと大きさを用いて、問合せと文書との類似度を計算
  - 実際には、一つの問合せベクトルに対して、各文書ベクトルの類似度計算が必要



# 文書のベクトル表現

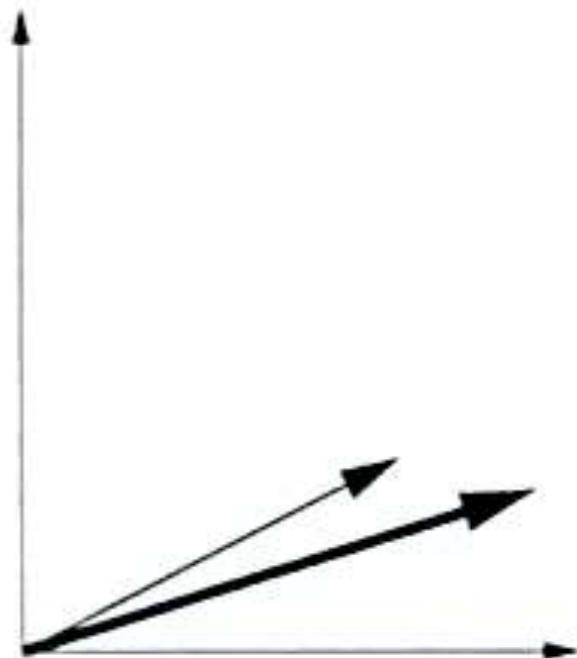


クライミングと山登りを軸とし、文書 3 と文書 4 がそれぞれベクトルにて表現されている。

上の例は2次元だが、実際の文書には多くの単語が含まれるため、多次元になる

これらの値に何を用いるかは後述

# ベクトルを用いた文書の類似度の比較



(a) 類似度が高い



(b) 類似度が低い



## 文書ベクトル

- 索引語をベクトルの成分とし、その索引語の重みを各成分の値とする
  - 索引語の  $m$  次元ベクトルとして表現
  - $m$  : 文書集合全体における索引語の異なり数
    - その文書だけではないことに注意
  - $j$  番目の文書  $d_j$  について、文書集合全体での  $i$  番目の索引語の文書  $d_j$  における重みを  $d_{ij}$  とすると、文書  $d_j$  のベクトルは右のように表現される

$$\mathbf{d}_j = \begin{bmatrix} d_{1j} \\ d_{2j} \\ \vdots \\ d_{mj} \end{bmatrix}$$



# 索引語-文書行列

- 文書集合  $D$  中に  $n$  個の文書ベクトルがある
  - 文書集合を、行が索引語、列が文書の行列で表現
  - $d_{ij}$  は索引語  $t_i$  の文書  $\mathbf{d}_j$  における重み
  - 重みは、ベクトル空間モデル自体では規定されない
  - 重みとして、たとえば以下のものが利用できる
    - 索引語の有無
    - 索引語の出現頻度
    - TF・IDF

$$D = \begin{bmatrix} d_{11} & d_{12} & \cdots & d_{1n} \\ d_{21} & d_{22} & \cdots & d_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ d_{m1} & d_{m2} & \cdots & d_{mn} \end{bmatrix}$$



# 索引語-文書行列の作成例

- 歌詞に「桜(さくら, サクラ, Sakura)」を含む, 以下の六つの楽曲の歌詞を文書として考える
- 歌詞1:「さくらさくら」(作者不明)
- 歌詞2:「SAKURAドロップス」(作詞:宇多田ヒカル)
- 歌詞3:「桜坂」(作詞:福山雅治)
- 歌詞4:「さくら」(作詞:ケツメイシ)
- 歌詞5:「桜」(作詞:小渕健太郎・黒田俊介)
- 歌詞6:「さくら(独唱)」(作詞:森山直太朗)

# 索引語-文書行列の例(表5.1)

ここでは、同義語  
や類義語を一つの  
索引語にまとめて  
いる

索引語の出現頻度

索引語 $t_i$		歌詞番号 $d_j$					
		$\vec{d}_1$	$\vec{d}_2$	$\vec{d}_3$	$\vec{d}_4$	$\vec{d}_5$	$\vec{d}_6$
		歌詞 1	歌詞 2	歌詞 3	歌詞 4	歌詞 5	歌詞 6
$t_1$	花 (花びら)	0	2	0	8	10	0
$t_2$	会う (会える)	0	0	0	0	2	2
$t_3$	君	0	2	7	12	3	3
$t_4$	今	0	0	5	0	0	3
$t_5$	咲かす (咲きほこる, 咲く)	0	2	0	0	2	0
$t_6$	桜 (さくら, サクラ, Sakura)	6	2	0	0	2	8
:	:						



## Bag of wordsモデル

- ベクトルによる表現では、文書中の単語の出現順序が考慮されない
- 「太郎 は 花子 より 賢い」と  
「花子 は 太郎 より 賢い」のベクトルは同じ
- これを**bag of words**モデルと呼ぶ
  - 「文書を単語ごとにバラバラにし、袋詰めにする」
- 構文解析などの高度な自然言語処理をせずに、  
単語の統計量だけを用いる手法
  - 文書分類などでも用いられる



## 問合せベクトル

- 問合せも文書と同様に、索引語を成分としたベクトルで表現
  - 次元数は文書ベクトルと同じく、文書集合全体における索引語の異なり数
  - 重みは、問合せに含まれる語は1、含まれない語は0とする
- たとえば、歌詞の索引語-文書行列で、問合せが「桜 AND 花」であれば、

$$q = \{1 \quad 0 \quad 0 \quad 0 \quad 0 \quad 1\}$$

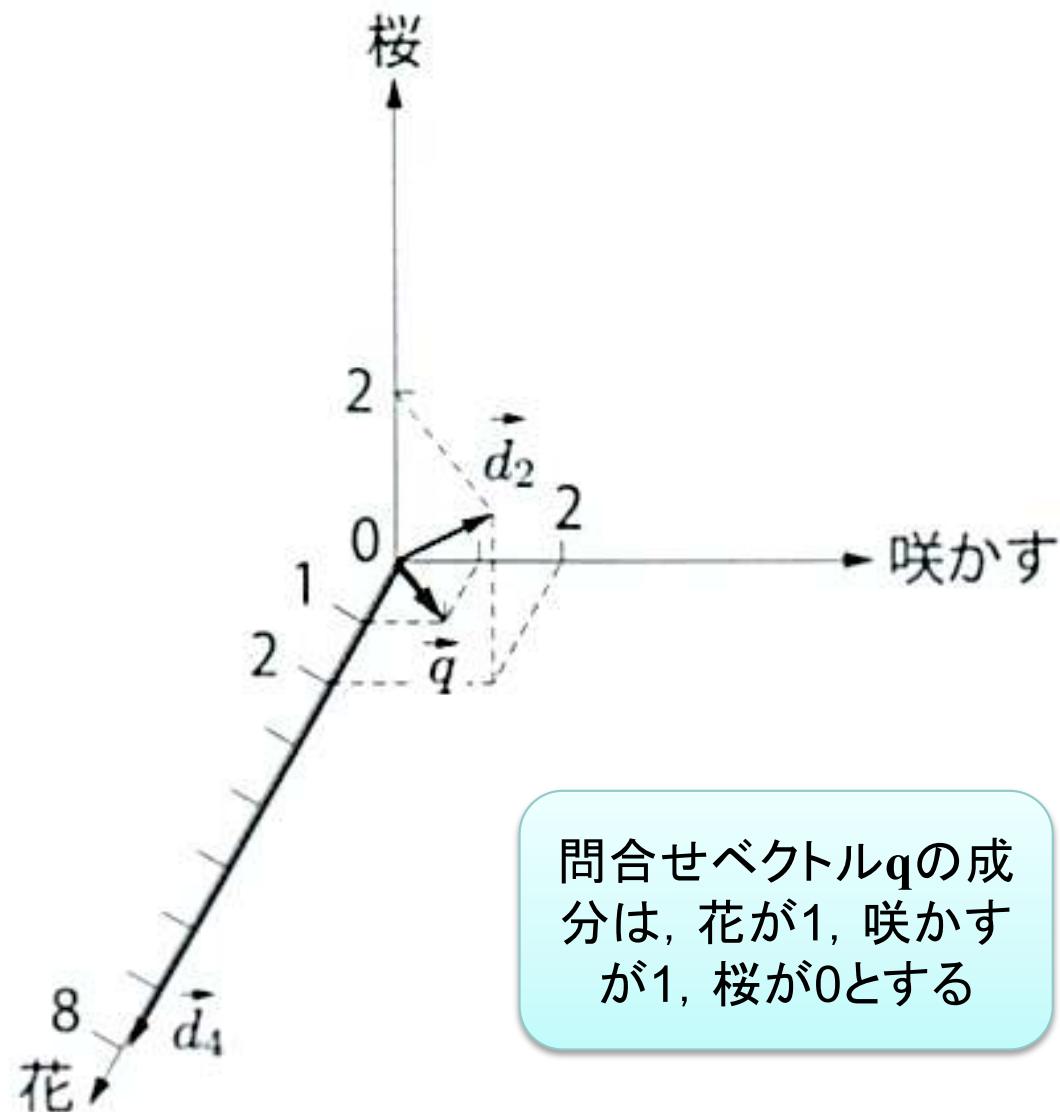
花 会う 君 今 咲かす 桜

ここで「AND」は  
ブール演算子では  
ないことに注意

$$\mathbf{q} = \begin{bmatrix} q_1 \\ q_2 \\ \vdots \\ q_m \end{bmatrix}$$



# ベクトル空間の例



- 「桜」「咲かす」「花」の三つの索引語を考える
- 以下の三つのベクトルは、左図で表せる
  - 歌詞2( $d_2$ )
  - 歌詞4( $d_4$ )
  - 問合せ( $q$ )



## ベクトルの類似度

- 二つのベクトルのなす角が小さいほど類似
- コサイン類似度

$$\cos(\mathbf{d}_j, \mathbf{q}) = \frac{\mathbf{d}_j \cdot \mathbf{q}}{\|\mathbf{d}_j\| \|\mathbf{q}\|} = \frac{\sum_{i=1}^m d_{ij} q_i}{\sqrt{\sum_{i=1}^m d_{ij}^2} \sqrt{\sum_{i=1}^m q_i^2}}$$

- 分子は二つのベクトルの内積
- 分母は二つのベクトルの大きさの積
- 値が大きいほど,  $\mathbf{d}_j$  と  $\mathbf{q}$  は類似している

# ベクトル空間モデルによる検索の例(1)

- 教科書の表5.1(p.45)に示した文書集合から、以下の索引語-文書行列が得られる

$$D = \begin{pmatrix} & d_1 & d_2 & d_3 & d_4 & d_5 & d_6 \\ t_1 & 0 & 2 & 0 & 6 & 10 & 0 \\ t_2 & 0 & 0 & 0 & 0 & 2 & 2 \\ t_3 & 0 & 2 & 7 & 12 & 3 & 3 \\ t_4 & 0 & 0 & 5 & 0 & 0 & 3 \\ t_5 & 0 & 2 & 0 & 0 & 2 & 0 \\ t_6 & 6 & 2 & 0 & 0 & 2 & 8 \end{pmatrix}$$



## ベクトル空間モデルによる検索の例(2)

- 問合せとして「花 AND 咲く」を考える

$$q = \{1 \quad 0 \quad 0 \quad 0 \quad 1 \quad 0\}$$

- 各文書のコサイン類似度を計算すると、以下のようになる

- $\cos(\mathbf{d}_1, q) =$

- $\cos(\mathbf{d}_2, q) =$

- $\cos(\mathbf{d}_3, q) =$

- $\cos(\mathbf{d}_4, q) =$

- $\cos(\mathbf{d}_5, q) =$

- $\cos(\mathbf{d}_6, q) =$



## $\cos(\mathbf{d}_2, \mathbf{q})$ の計算例

$$\mathbf{d}_2 = \begin{matrix} t_1 & t_2 & t_3 & t_4 & t_5 & t_6 \\ \{2 & 0 & 2 & 0 & 2 & 2\} \end{matrix}$$

$$\mathbf{q} = \begin{matrix} t_1 & t_2 & t_3 & t_4 & t_5 & t_6 \\ \{1 & 0 & 0 & 0 & 1 & 0\} \end{matrix}$$

$$\begin{aligned}\cos(\mathbf{d}_2, \mathbf{q}) &= \frac{2 \cdot 1 + 0 \cdot 0 + 2 \cdot 0 + 0 \cdot 0 + 2 \cdot 1 + 2 \cdot 0}{\sqrt{2^2 + 0^2 + 2^2 + 0^2 + 2^2 + 2^2} \sqrt{1^2 + 0^2 + 0^2 + 0^2 + 1^2 + 0^2}} \\ &= \frac{4}{\sqrt{16} \sqrt{2}} \\ &= \frac{1}{\sqrt{2}} \\ &\cong 0.707\end{aligned}$$



# ベクトル空間モデルの利点

## (1)

- 問合せを完全に含んではいないが、利用者の問合せに適している可能性が高い文書を検索可能
  - 問合せと文書の類似度を用いているため
  - 前の例では  $d_4$  が相当
- 類似度の値を用いてランキングすることができる
  - 前の例では、以下のようにランキングできる
    - 第1位:  $d_5$
    - 第2位:  $d_2$
    - 第3位:  $d_4$

# ベクトル空間モデルの利点 (2)

- 問合せ中の検索語に重みを与えることが可能
  - 前の例では、各検索語の重みをすべて1としていた



- より「咲く」という語を重視したければ、 $q$  中の  $t_5$  の値を2, 3, ...のように増やせば良い
- 検索結果を見て、問合せベクトル中の索引語の重みを修正し、再度検索できる
  - これを用いて検索結果の向上を図る手法を適合性フィードバックと呼ぶ（次回説明）



# ベクトル空間モデルの欠点

## ○ 計算量の問題

- 問合せベクトルとすべての文書ベクトルとの類似度を計算する必要
- ブーリアンモデルより計算量が大きい
- ただし、問合せベクトル中の索引語と文書中の索引語がまったく重複しない場合は計算不要
  - ・ 計算しなくても、類似度が0になることは明らか



# 確率モデルとは

- 確率論に基づいて問合せに対する文書の適合度を求めるモデル
- ある文書  $d$  が問合せに適合する(事象  $R$ ) 確率  $P(R|d)$  と適合しない確率  $P(\bar{R}|d)$  の比  $g(d)$  によって適合度を計算

$$g(d) = \log \frac{P(R|d)}{P(\bar{R}|d)}$$

$g(d)$ が正なら適合, 負なら不適合

- 文書集合の個々の索引語の出現は独立と仮定
- 文書中に出現する語としない語の両方の情報を用いる

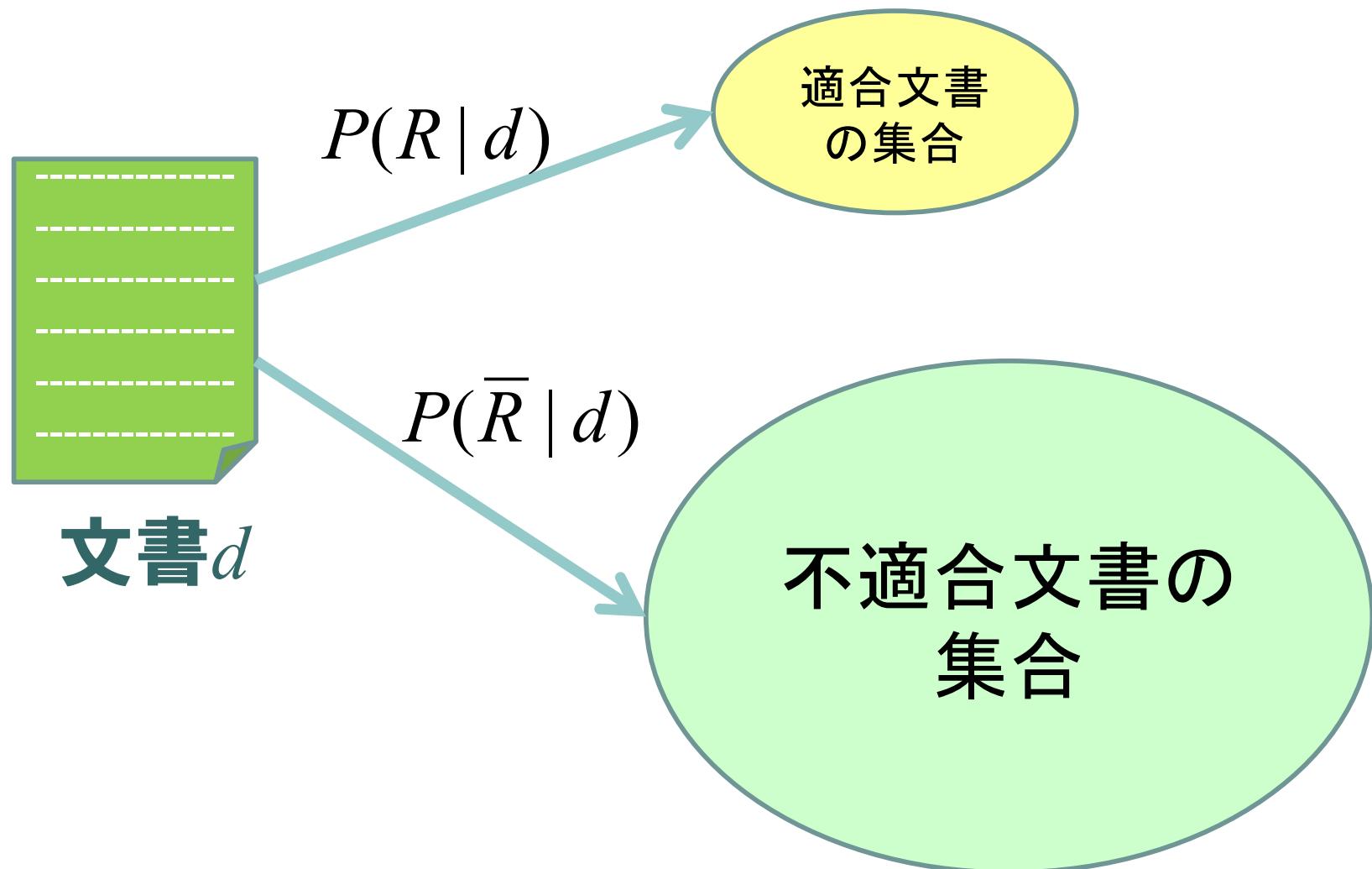
## 条件付き確率

事象  $A$  が起こったとき事象  $B$  が起きる確率

$$P(B|A) = \frac{P(A \cap B)}{P(A)}$$



図で表すと. . .





# 確率モデルの定式化(1)

- ベイズの定理を適用

$$g(d) = \log \frac{P(d | R)}{P(d | \bar{R})} + \log \frac{P(R)}{P(\bar{R})}$$

定数となるため、以  
降の計算では省略

ベイズの定理

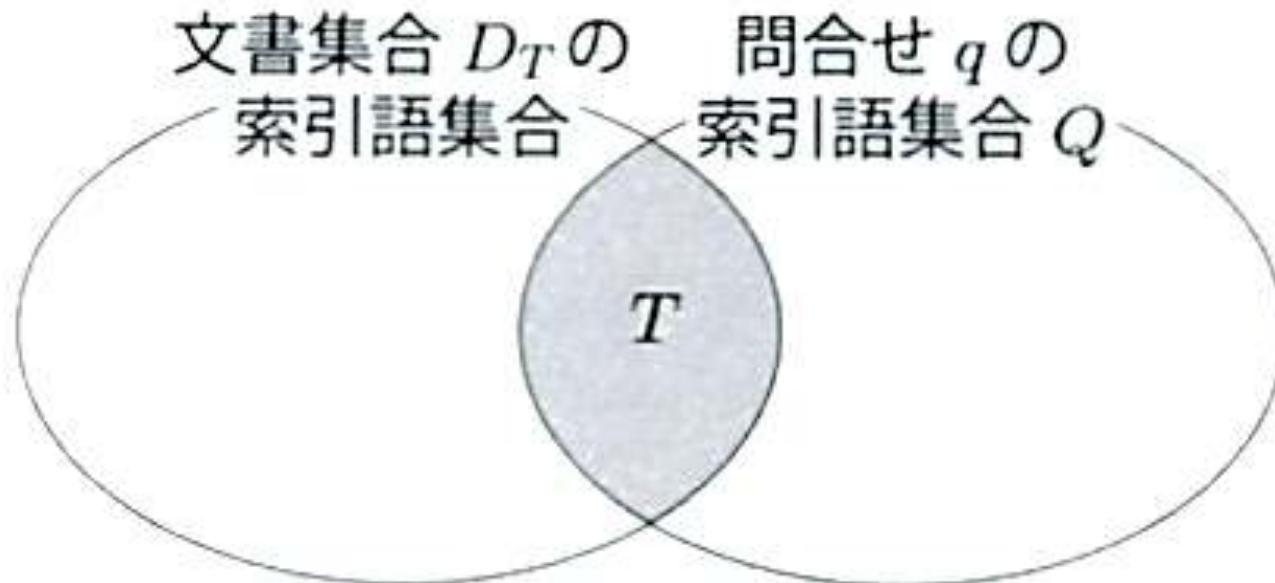
$$P(A | B) = \frac{P(A)P(B | A)}{P(B)}$$

- ここで、問合せ  $q$  に含まれる索引語の集合  $Q$  と、その部  
分集合  $T$ を考える
- このとき、文書中に含まれる索引語の集合と  $Q$  の積集  
合が  $T$  に等しくなるような文書集合を  $D_T$  で表す
  - $D_T$ に含まれる文書は、 $T$ の要素である索引語を含み、  
 $Q - T$ の要素である索引語を含まない



## 索引語集合 $T$ の意味

- 文書中に含まれる索引語の集合と、問合せに含まれる索引語の集合  $Q$  の積集合が  $T$  になる文書集合を  $D_T$  で表す





## 確率モデルの定式化(2)

- 文書  $d$  が  $D_T$  に属するとすると、

$$P(d | R) = \prod_{t_i \in T} P(X_i = 1 | R) \prod_{t_j \in Q-T} P(X_j = 0 | R)$$

$X_i$  は、索引語  $t_i$  が文書  $d$  中に出現する( $X_i=1$ )、あるいは、  
しない( $X_i=0$ )事象を表す確率変数

- 上の式は、文書が適合の場合に、 $T$  のみを含む文書を生成する確率を示す



## 確率モデルの定式化(3)

- $\bar{R}$  も同様に求められる

$$P(d \mid \bar{R}) = \prod_{t_i \in T} P(X_i = 1 \mid \bar{R}) \prod_{t_j \in Q-T} P(X_j = 0 \mid \bar{R})$$

$X_i$  は、索引語  $t_i$  が文書  $d$  中に出現する ( $X_i=1$ )、あるいは、  
しない ( $X_i=0$ ) 事象を表す確率変数

- 上の式は、文書が不適合の場合に、 $T$  のみを  
含む文書を生成する確率を示す



## 確率モデルの定式化(4)

- 前ページの  $P(d|R)$  と  $P(d|\bar{R})$  を  $g(d)$  の式に代入

$$g(d) = \sum_{t_i \in T} \log \frac{P(X_i = 1 | R)}{P(X_i = 1 | \bar{R})} + \sum_{t_j \in Q-T} \log \frac{P(X_j = 0 | R)}{P(X_j = 0 | \bar{R})}$$

- $P(X_i = 1 | R)$  を  $p_i$  (適合文書に索引語  $t_i$  が付与されている確率),  $P(X_i = 1 | \bar{R})$  を  $\bar{p}_i$  (不適合文書に索引語  $t_i$  が付与されている確率) で表す

$$g(d) = \sum_{t_i \in T} \log \frac{p_i}{\bar{p}_i} + \sum_{t_j \in Q-T} \log \frac{1-p_i}{1-\bar{p}_i}$$



## 確率モデルの定式化(5)

$$g(d) = \sum_{t_i \in T} \log \frac{p_i}{\bar{p}_i} + \sum_{t_j \in Q-T} \log \frac{1-p_i}{1-\bar{p}_i}$$

- 上式を、索引語に対する重み付けという観点から解釈すると…
  - $\log p_i / \bar{p}_i$  は、問合せと文書中の両方に出現する索引語( $T$ )の重み
  - $\log(1-p_i)/(1-\bar{p}_i)$  は、問合せに出現するが文書中には出現しない索引語( $Q-T$ )の重み
- 上記  $g(d)$  によって、問合せに対する文書の適合度を計算できる



# 確率パラメータの推定(1)

- 確率  $p_i$  や  $\bar{p}_i$  はどうやって求める?
  - 適合文書がすべてわかっていれば,

	適合文書数	不適合文書数	文書数
$t$ が付与されている	$r$	$n-r$	$n$
$t$ が付与されていない	$N_r-r$	$N-N_r-(n-r)$	$N-n$
合計	$N_r$	$N-N_r$	$N$

$N$  :全文書数

$N_r$  :問合せに適合する文書数

$n$  :索引語  $t$  が付与されている文書数

$r$  :索引語  $t$  が付与されている適合文書数

最尤推定により,

$$p_i = \frac{r}{N_r}, \bar{p}_i = \frac{n-r}{N-N_r}$$



## 確率パラメータの推定(2)

- 実際は、問合せに対する適合文書はわからない
  - わかっていれば検索する必要がない
- 最初は適当な初期値を用い、利用者による適合性の判断でパラメータを更新
  - $p_i$  の初期値は 0.5,  $\bar{p}_i$  の初期値は  $df(t_i) / N$ 
    - 適合文書に索引語  $t_i$  が付与されるかどうかは不明
    - 不適合文書に索引語  $t_i$  が付与される確率は、その語がどれくらいの割合の文書に含まれるかに比例
  - 利用者による適合・不適合の判断の繰り返しにより、パラメータの精度が上がる



# 確率モデルの利点・欠点

## ○ 利点:

- 理論的基礎がしっかりとしており、値の意味も明確
- 適合・不適合の判定を繰り返すことで、パラメータの精度が改善

## ○ 欠点:

- 確率パラメータの設定の問題



## 検索モデルのまとめ

- 情報検索のモデルの代表的なものとして、ブーリアンモデル、ベクトル空間モデル、確率モデルがある
- ブーリアンモデルは計算量が少ない
  - ただし、ランキングができない
- ベクトル空間モデルと確率モデルはランキングが可能
  - 問合せとすべての文書との類似度・適合度を計算する必要があるため、計算量が大きい