

Boolean модел на извличане на информацията

проф. д-р инж. Христо Вълчанов

<http://cs.tu-varna.bg>

Извличане на информация

- Системи за управление на бази данни – организация на данните.
- Системи за извличане на информация – неструктурирани данни

Системи за управление на бази данни

```
SELECT Name, Bank, Transaction FROM  
Clients WHERE Amount > 100000
```

Системи за извличане на информация

Няма информация за структурата на документите.

Запитване:

“Достави имената на всички документи, съдържащи изрази с думите ‘банка’ и ‘трансфер’ “

Системи за управление на бази данни

Оптимизация на запитванията:

- Сортиране на данните;
- Алгоритми за търсене;
- Промяна регистъра на буквите.

Системи за извличане на информация

Не могат да променят текста в документите.

Boolean модел на извличане на информацията

Използват се булеви изрази в запитванията.

Определя се кои документи съдържат или не определено множество от думи.

Boolean изрази

- **AND;**
- **OR;**
- **NOT;**
- **NEAR;**
- **WITH;**
- **маски.**

Boolean изрази - AND

Реализира операцията *конюнкция*.

Търсят се всички документи, които съдържат всички указани думи.

bank AND transaction

Boolean изрази - OR

Реализира булевата операция *дизюнкция*.

Търсят се всички документи, които
съдържат някоя или всички указани думи.

bank OR transaction

Boolean изрази - NOT

Реализира булевата операция *отрицание*.

Търсят се всички документи, които
съдържат някоя но не и друга указана дума.

bank AND NOT transaction

Boolean изрази - NEAR

Изисква наличието на дума, която е близко на n позиции от указаната.

bank NEAR 2 transaction

Boolean изрази - WITH

Изисква наличието на дума, която се използва съвместно с друга.

bank WITH transaction

Boolean изрази - маски

Указва съвпадение с произволни символи..

comput*

Реализация на булеви запитания

- Чрез матрица на съответствията;
- Чрез инвертен индекс.

Матрица на съответствията

- Всеки ред съответства на отделна дума;
- Всяка колона съответства на отделен документ;
- С всяка дума е свързано тегло 0 или 1.

Пример

d1: jaguar₂ new₅ world₆ mammal₇ felidae₁₀ family₁₁

d2: jaguar₁ design₃ four₄ new₅ engine₆

d3: jaguar₂ atari₃ keen₅ 68k₉ family₁₀ device₁₁

d4: jacksonville₂ jaguars₃ professional₆ us₇ football₈ team₉

**d5: mac₁ os₂ x₃ jaguar₄ available₆ price₉ us₁₁ \$199₁₂ apple₁₄ new₁₅
family₁₆ pack₁₇**

**d6: one₁ such₂ rule₃ family₄ incorporate₆ jaguar₈ their₁₀ name₁₁ jaguar₁₃
paw₁₄**

d7: big₄ cat₅

Матрица на съответствията

family d_1, d_3, d_5, d_6
football d_4
jaguar $d_1, d_2, d_3, d_4, d_5, d_6$
new d_1, d_2, d_5
rule d_6
us d_4, d_5
world d_1

...

	d_1	d_2	d_3	d_4	d_5	d_6
family	1	0	1	0	1	1
football	0	0	0	1	0	0
jaguar	1	1	1	1	1	1
new	1	1	0	0	1	0
us	0	0	0	1	1	0
world	1	0	0	0	0	0
...						

Изпълнение на заявка

Изпълнението на заявката се състои в извличане на редовете, съответстващи на указаните в заявката думи, и прилагане на булевите операции върху тях.

Изпълнение на заявка

jaguar AND family

	d₁	d₂	d₃	d₄	d₅	d₆
family	1	0	1	0	1	1
jaguar	1	1	1	1	1	1
	1	0	1	0	1	1

Резултат: d₁, d₃, d₅, d₆

Изпълнение на заявка

football OR world

	d_1	d_2	d_3	d_4	d_5	d_6
football	0	0	0	1	0	0
world	1	0	0	0	0	0
	1	0	0	1	0	0

Резултат: d_1, d_4

Изпълнение на заявка

jaguar AND NOT football

	d_1	d_2	d_3	d_4	d_5	d_6
football	0	0	0	1	0	0
	1	1	1	0	1	1
jaguar	1	1	1	1	1	1
	1	1	1	0	1	1

Резултат: d_1, d_2, d_3, d_5, d_6

Изпълнение на заявка

jaguar AND football AND NOT new

	d₁	d₂	d₃	d₄	d₅	d₆
new	1	1	0	0	1	0
	0	0	1	1	0	1
football	0	0	0	1	0	0
jaguar	1	1	1	1	1	1
	0	0	0	1	0	0

Резултат: d₄

Матрица на съответствия - недостатъци

- Размер на матрицата;
- Удобна за прости булеви операции;
- Няма възможност за рейтинговане на резултантното множество.

Реализация на булеви запитвания чрез инвертен индекс

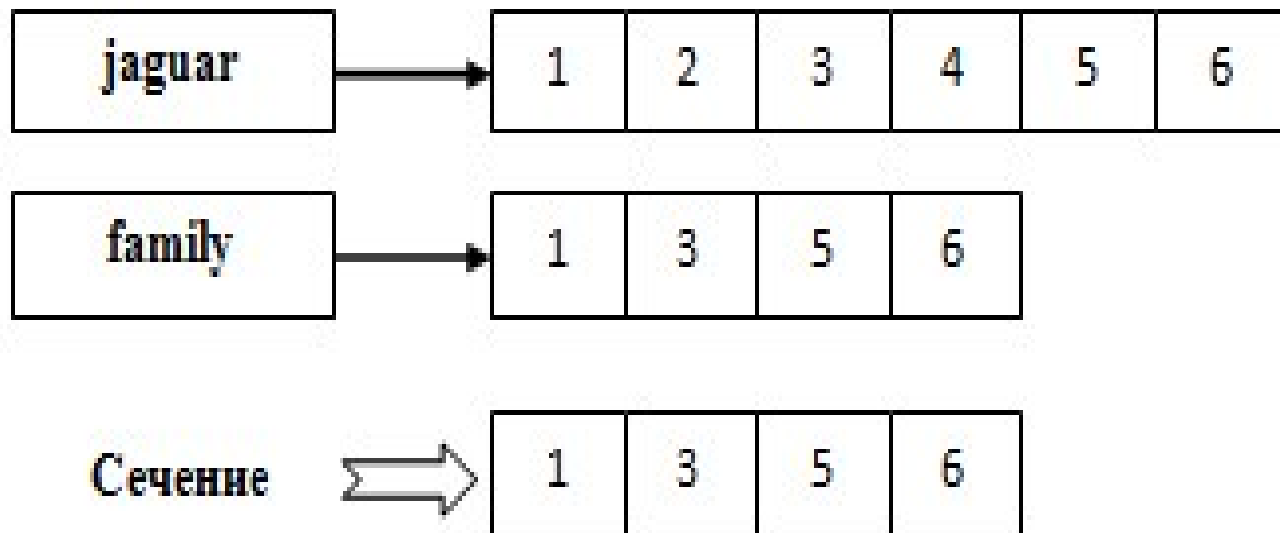
- Използване на постинг списъците на указани думи;
- Манипулиране с постинг списъците.

Пример

jaguar AND family

1. Намира се думата **jaguar** от речника.
2. Извлича се нейния постинг списък.
3. Намира се думата **family** от речника.
4. Извлича се нейния постинг списък.
5. Намира се сечението на двата постинг списъка.

Пример



Алгоритъм на сливане на списъци - merging

```
INTERSECT(p1, p2)
  answer <- {}
  while p1 != NIL and p2 != NIL
  do if docID(p1) = docID(p2)
    then ADD(answer, docID(p1))
    p1 <- next(p1)
    p2 <- next(p2)
  else if docID(p1) < docID(p2)
    then p1 <- next(p1)
  else p2 <- next(p2)
  return answer
```

Оптимизация на реализация на запитванията

*Важен е редът, в който постинг
списъците се обработват.*

Оптимизация

Стандартен евристичен подход:

Обработване на думите в нарастваща последователност на тяхното появяване в документите (честота на появяване).

Оптимизация – пример1

jaguar AND family AND world

family - 4

football - 8

jaguar - 6

new - 3

rule - 1

us - 2

world - 1

...

$d_1/11, d_3/10, d_5/16, d_6/4$

$d_4/8$

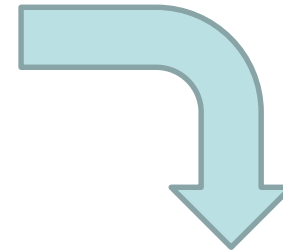
$d_1/2, d_2/1, d_3/2, d_4/3, d_5/4, d_6/8$

$d_1/5, d_2/5, d_5/15$

$d_6/3$

$d_4/7, d_5/11$

$d_1/6$



(world AND family) AND jaguar

Оптимизация – пример2

**(phone OR tablet) AND (android OR windows)
AND (black OR white)**

- Оценява се размера на всеки OR като сума от честотите на неговите компоненти;
- Заявката се обработва в нарастваща последователност на размера на всеки дизюнктивен терм.

Въпроси?