Úvod do dokumentografických databází a systémů (DIS)

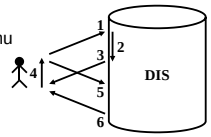
Přehled problematiky

Vznik DIS

- **50. léta 20. stol.**
- **Postupná automatizace postupů používaných v knihovnictví**
- **Nyní samostatná podčást IS**
  - **Faktografický IS**
    - **Zpracování informací s definovanou vnitřní strukturou (nejčastěji v podobě tabulek)**
  - **Dokumentografický IS**
    - Zpracování informací v podobě textu v přirozeném jazyce bez pevné vnitřní struktury

Práce s DIS

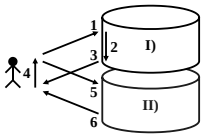
1. Zadání dotazu
2. Porovnání
3. Získání seznamu odpovídajících dokumentů
4. Ladění dotazu
5. Vyžádání dokumentu
6. Obdržení textu



Struktura DIS

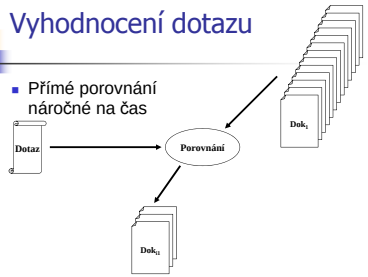
I) Systém zpřístupnění dokumentů
- Vrací sekundární informace
  - Autor
  - Název
  - ...

II) Systém dodání dokumentů
- Někdy není řešen pomocí SW



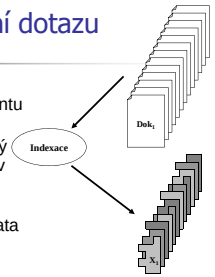
Vyhodnocení dotazu

- Přímé porovnání náročné na čas



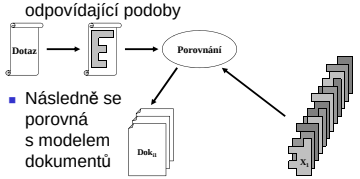
Vyhodnocení dotazu

- Nutné vytvoření modelu dokumentu
- Ztrátový proces, obvykle založený identifikaci slov v dokumentech
- Výsledkem strukturovaná data vhodná pro porovnávání



Vyhodnocení dotazu

- Dotaz se upraví do odpovídající podoby
- Následně se porovná s modelem dokumentů



Předzpracování textu

- Vyhledávání probíhá nad vytvořeným modelem efektivněji, ale může použít jen informace obsažené v modelu.
- Cílem je vytvořit model, který by zachoval co nejvíce informací, obsažených v původním modelu.
- Problémem je řada nejednoznačností.
- Dosud neřešitelné nároky na encyklopedické i asociativní znalosti.

Porozumění textu

- Homonymie slov
  - Jedno slovo může používat stejný tvar pro různé pády a další gramatické jevy (gramatická homonymie)
    - **kontroly**: 1. p. m.č., 2. p. j.č.
      není zřejmé, zda se jedná o jednu, nebo více kontrol
  - Jeden tvar slova může mít různý význam
    - **plesy**: podst. jm. *ples*, podst. jm. *pleso*
    - **žena**: podst. jm. *žena*, sloveso *hnát*
    - **hnát**: sloveso *hnát*, podst. jm. *hnát*
    - **tři**: číslovka *tři*, sloveso *třít*
    - **pět**: číslovka *pět*, sloveso *pět*

Porozumění textu

- Jednotlivá přiřazení jsou navíc závislá na subjektu, který dokument píše nebo čte.
  - Dva lidé mohou jednomu slovu přikládat zcela nebo jen částečně jiný význam.
  - Dva lidé si i pod stejným významem mohou představit jiný konkrétní předmět nebo množinu předmětů.
    - **máma**, **pokoj**, ...
- Výsledkem je situace, kdy dva různí čtenáři nemusí přečtením získat stejnou informaci jako autor, ani navzájem.

Porozumění textu

- Homonymie a nejednoznačnosti narůstají při přechodu od slov k větám.
  - Homonymie vlastních jmen na začátku věty
    - *Dohnal* zvítězil. (*Čtrnáctý* zvítězil.)
    - *Dohnal* předešel gen. *Kvapila* velmi výrazně. - jedna, nebo dvě věty?
  - Homonymie spojky *a* v předmětu věty
    - Funkce rezistoru a zesilovače v radiotechnice.
      (funkce rezistoru v radiotechnice) a (funkce zesilovače v radiotechnice)
      (funkce rezistoru) a (funkce zesilovače v radiotechnice)
      (funkce rezistoru) a (zesilovače v radiotechnice)
  - Homonymie podmětu a předmětu
    - *Popílek* přikryl *sníh*. – co leží navrchu?

Přesnost a úplnost

- Výsledkem nejednoznačností žádný existující DIS nedává ideální výsledky
- Po zobrazení odpovědi na dotaz lze určit následující
  - Počet vrácených dokumentů **Nv**
    - O nich si DB myslí, že jsou relevantní, odpovídají dotazu
  - Počet vrácených relevantních dok. **Nvr**
    - O nich si tazatel myslí, že uspokojují jeho požadavky
  - Počet všech relevantních dok. v DB **Nr**
    - Problematické u velkých DB

Přesnost a úplnost

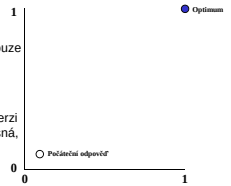
- Kvalita výsledné množiny dokumentů se měří na základě těchto čísel
  - Přesnost (Precision)
    - **P = Nvr / Nv**
    - Pravděpodobnost, že dokument zařazený v odpovědi je skutečně relevantní
  - Úplnost (Recall)
    - **R = Nvr / Nr**
    - Pravděpodobnost, že skutečně relevantní dokument je zařazený v odpovědi

Přesnost a úplnost

- Koeficienty jsou opět závislé na subjektivním názoru tazatele
- Dokument vrácený na výstupu může uspokojovat požadavky dvou uživatel, kteří položili stejný dotaz, různou měrou.

Přesnost a úplnost

- V ideálním případě
  - **P=R=1**
  - V odpovědi jsou zařazeny právě a pouze všechny relevantní dokumenty
- V běžném případě
  - Odpověď na první verzi dotazu není ani přesná, ani úplná



Kritérium predikce

- Při formulaci dotazů je potřebné uhádnout, které termy (slova) byly v dokumentu autorem použity pro vyjádření dané myšlenky
  - Problémy m.j. způsobují
    - Synonyma (autor mohl použít synonymum, které si tazatel při formulaci dotazů ani nemusí neuvědomi)
    - Překrývající se významy slov
    - Opisy jedné situace jinými slovy

Kritérium predikce

- Částečným řešením je zařazení tezauru, který obsahuje
  - Hierarchie slov a jejich významů
  - Synonyma slov
  - Asociace mezi slovy
- Tazatel může tezaurus využít při formulaci svých dotazů

Kritérium predikce

- Při ladění dotazů má uživatel tendenci postupovat konzervativně
  - V dotazu zůstávají často ty jeho části, které uživatele napadly na začátku a mění se jen podružné části, které nekvalitní výsledek nemusí nijak zásadně ovlivnit
- Vhodné je uživateli pomoci s odstraněním nevhodných částí dotazu, které nepopisují relevantní dokumenty a naopak s přidáváním formulací, které relevantní dokumenty popisují

Kritérium maxima

- Tazatel obvykle není schopen (nebo ochoten) procházet příliš mnoho dokumentů do té míry, aby se rozhodl, zda jsou pro něj relevantní nebo ne
- Obvykle 20-50 podle velikosti

⇒ Potřeba nejen dokumenty rozlišovat na odpovídající/neodpovídající dotazu, ale řadit je na výstupu podle míry předpokládané relevance

Kritérium maxima

- V důsledku kritéria maxima se při ladění dotazu uživatel obvykle snaží zvýšit přesnost
  - Malé možství dokumentů v odpovědi, obsahující co největší poměr relevantních dokumentů
- Některé oblasti použití vyžadují co nejvyšší přesnost i úplnost
  - Právnictví

Modely

- Úrovně modelů
  - Rozlišují (ne)přítomnost slov v dokumentech
  - Rozlišují frekvence výskytů slov
  - Rozlišijí pozice výskytů slov v dokumentech

Boolský model DIS

- 50-tá léta 20. stol.
- Automatizace postupů používaných v knihovnictví

Boolský model DIS

- Databáze **D** obsahující **n** dokumentů
  - **D**=$\{d_1, d_2, \ldots d_n\}$
- Dokumenty popisovány pomocí **m** termů
  - **T** =$\{t_1, t_2, \ldots t_m\}$
  - term t_j = slovo nebo sousloví
- Reprezentace dokumentu pomocí množiny termů
  - Obsažených v dokumentu
  - Popisujících význam dokumentu
  - $d_1 \subseteq$ **T**

Boolský model DIS

- Databáze **D** obsahující **n** dokumentů
  - **D**=$\{d_1, d_2, \ldots d_n\}$
- Dokumenty popisovány pomocí **m** termů
  - **T** =$\{t_1, t_2, \ldots t_m\}$
  - term t_j = slovo nebo sousloví
- Reprezentace dokumentu pomocí množiny termů
  - Obsažených v dokumentu
  - Popisujících význam dokumentu
  - $d_1 \subseteq$ **T**

Indexace Boolského DIS

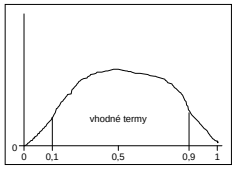
- Přiřazení množiny termů, které jej popisují ke každému dokumentu
  - Ruční
    - Nekonzistence
  - Automatická
    - Konzistentní, ale bez porozumění textu
  - Řízená
    - Předem daná množina termů
  - Neřízená
    - Množina termů se mění s přibývajícími dokumenty

Indexace

- Tezaurus
  - Vnitřně strukturovaná množina termů
    - Synonyma s preferovanými termy
    - Hierarchie užších/širších termů
    - Příbuzné termy
    - ...
- Stop-list
  - Nevýznamová slova

Indexace

- Příliš obecná slova nejsou pro identifikaci dokumentů vhodná
- Příliš specifická slova také ne



Boolský model DIS

- Dotaz je vyjádřený logickým výrazem
  - t_a **AND** t_b v dokumentu se vyskytují oba termy
  - t_a **OR** t_b v dokumentu se vyskytuje alespoň jeden z termů
  - **NOT** t v dokumentu se daný term nevyskytuje

Boolský model DIS

- Dotazem tedy může být například:
  - 'vyhledávání' **AND** 'informace'
  - 'kódování' **OR** 'dekódování'
  - 'zpracování' **AND** ('dokument' **OR** 'text')
  - 'počítač' **AND NOT** 'osobní'

Boolský model DIS

- Víceslovné termy v dotazech
  - 'vyhledávání informace'
  - 'kódování dat' **OR** 'dekódování dat'
  - 'zpracování textu'
  - 'počítač' **AND NOT** 'osobní počítač'

Organizace indexu

- Invertovaný seznam
  - Pro každý term seznam dokumentů ve kterých se vyskytuje
    - $t_1 = d_{1,1}, \quad d_{1,2}, \quad \ldots, \quad d_{1,k1}$
    - $t_2 = d_{2,1}, \quad d_{2,2}, \quad \ldots, \quad d_{2,k2}$
    - $t_m = d_{m,1}, \quad d_{m,2}, \quad \ldots, \quad d_{m,km}$

Organizace indexu

- Zpracování dokumentů na vstupu
  - vznikne posloupnost dvojic <dok_id,term_id> setříděná dle v uvedeném pořadí
- Setřídění dle term_id,dok_id

Nevýhody Boolského DIS

- Salton:
  - Formulace dotazů je spíše uměním než vědou.
  - Nemožnost ohodnotit vhodnost vystupujících dokumentů.
  - Všechny termy v dotazu i v identifikaci dokumentu jsou chápány jako stejně důležité.
  - Neumožnost řízení velikosti výstupu.
  - Některé výsledky neodpovídají intuitivní představě.
    - V disjunktivním dotazu na výstupu dokumenty obsahující jediný z termů vedle dokumentů obsahujících všechny.
    - V konjunktivním dotazu na výstupu nejsou dokumenty neobsahující žádný z termů ani dokumentů neobsahující jeden z nich.

Vektorový model DIS

- 70-tá léta 20. stol.
  - cca o 20 let mladší než Boolské DIS
- Snaha minimalizovat nebo odstranit nevýhody Boolských DIS

Vektorový model DIS

- Databáze **D** obsahující **n** dokumentů
  - **D**={d_1, d_2, … d_n}
- Dokumenty popisovány pomocí **m** termů
  - **T** ={t_1, t_2, … t_m}
  - term t_j = slovo nebo sousloví
- Reprezentace dokumentu pomocí vektoru vah termů

$$\vec{d_i}=<w_{i,1},w_{i,2},\ldots,w_{i,m}>$$

Aktuální možnosti zpracování textu

Fulltextové vyhledávání

Filtrace, Disambiguace, Lemmatizace, Indexy, Tezaury, Dotazování

Fulltextové vyhledávání

- Odlišné od principů běžného vyhledávání
  - Neprohledávají se striktně strukturovaná data, kde má každý sloupec každé tabulky předem daný význam
  - Prohledávají se volně psané texty, kde může být stejná událost popsaná více autory rozdílně
    - Různá slova stejného významu (Synonyma)
    - Různé slovní obraty a opisy
    - …

Fulltextové vyhledávání

- Databázové systémy využívají svých prostředků rozšiřitelnosti a dodávají standardně prostředky, které vyhledávání v textových datech umožňují

Fulltextové vyhledávání

- Rozdílné přístupy a možnosti
  - Neexistuje objektivně nejlepší řešení
  - Výsledky navíc podléhají subjektivním názorům tazatelů

Fulltextové vyhledávání

- Samotná formulace dotazu, který by vrátil všechny dokumenty, které tazatele zajímají a žádné jiné obvykle nelze zformulovat
  - Spolu s vyhovujícími – relevantními – odpověďmi se obvykle vrací i odpovědi nerelevantní

Fulltextové vyhledávání

- Problémy
  - Homonyma
    - Ptá se tazatel dotazem „koruna“ na finanční, lesnické či panovnické dokumenty?
  - Synonyma
    - Vyhovuje dokument o „krychlích“ dotazu na dokumenty o „kostkách“?
    - Vyhovuje dokument o „stromech“ dotazu na „souvislé grafy bez cyklů“?

Fulltextové vyhledávání

- Problémy
  - Hierarchie významů
    - Zvíře – Savec – Šelma – Medvěd
    - Tiskovina – Časopis
  - Ohebnost slov
    - Jít, Jde, Jdu, Jdou, …

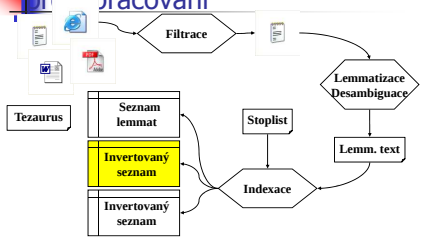
Fulltextové vyhledávání

- Striktní boolská logika není pro formulaci dotazů příliš vhodná
  - Dokument buďto vyhovuje dotazu, nebo nevyhovuje
  - Dotazování v textech vyžaduje třídit odpovědi podle předpokládané vhodnosti pro tazatele
    - Je potřebné mít možnost definovat míru shody dotazu s dokumentem

Obvyklý postup předzpracování

- Databáze obvykle používají některý z boolských modelů reprezentace dokumentů
  - Nejlépe odpovídá běžným dotazům
  - Relativně snadno se implementuje
  - Dotazy jsou ve formě boolských formulí, ve kterých operandy tvoří jednotlivá slova
    - Řada různých modifikací

Obvyklý postup předzpracování



Obvyklý postup předzpracování

- Filtrace
  - Odstraní formátovací značky a nechá čistý ASCII text
- Desambiguace
  - Určí význam slova podle kontextu
    - „pět chvály“ … sloveso pět
    - „pět vozidel“ … číslovka pět
- Lemmatizace
  - Určí základní tvar slova a gramatický tvar v dokumentu
  - Často nahrazen pomocí stemmeru, který hledá kmen slova

Obvyklý postup předzpracování

- Indexace
  - Vytvoří pomocné seznamy lemmat a dokumentů a invertovaný soubor
    - dvojice [*id_dok,id_lemmatu*] setříděné dle id_lemmatu a zbavené duplicit
    - dnes obvykle více informací, např. pětice [*id_dok,č_odstavce,č_věty,č_slova,id_lemmatu*] setříděné id_lemmatu
      - Dovoluje vyhodnocování tzv. proximitních omezení na vzdálenost slov v dokumentu

Fulltextové vyhledávání

- Pro podporu vyhledávání je potřeba nad textovým sloupcem vytvořit index – invertovaný soubor
- Běžné textové sloupce jsou pro tyto účely krátké a nevyhovující
  - Obvykle se takto indexují sloupce některého z LOB (Large OBject) typů

Large OBjects (LOB)

- LOBy
  - Standardní typy pro ukládání objemných dat na serveru
    definován v SQL-92 Full
    - Až 4GB dat
      - BLOB … standardní binární typ
      - CLOB … znakový typ v univerzální znakové sadě serveru
      - NCLOB … znakový typ v národní znakové sadě serveru
  - V Oracle navíc externí typ BFILE
    - pouze pro čtení
    - samostatný binární soubor uložený **vně databáze** v OS

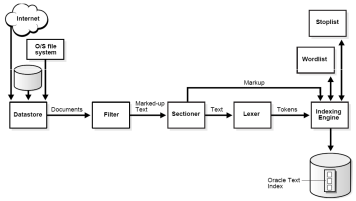
Large OBjects (LOB)

- LOBy
  - Standardní typy pro ukládání objemných dat na serveru
    definován v SQL-92 Full
    - Až 4GB dat
      - BLOB … standardní binární typ
      - CLOB … znakový typ v univerzální znakové sadě serveru
      - NCLOB … znakový typ v národní znakové sadě serveru
  - V MS SQL
    - Image … binární data do velikosti 2 GB
    - Text … textová data do velikosti 2 GB
    - NText … textová data v národní znakové sadě do vel. 1 GB

Large Objects

- Ve sloupcích tabulky je uložen pouze deskriptor (tzv. LOB lokátor), odkazující na samotně uložená data

Struktura Oracle full-text



Oracle fulltext

- Filtrování vstupních dokumentů
  - NULL_FILTER pro textové dokumenty
    - TXT, HTML, XML
  - INSO_FILTER pro binární dokumenty
  - CHARSET_FILTER pro konverzi získaných dokumentů do znakové sady databáze

Oracle fulltext

- Druhy fulltextových indexů
  - CONTEXT
    - Základní typ indexu pro vyhledávání v textových datech
    - Vhodný pro větší dokumenty
    - Synchronizace indexu s daty je nutné provést explicitně zavoláním CTX_DDL.SYNC_INDEX (obdoba WITH CHANGE_TRACKING MANUAL z MS SQL)

Oracle fulltext

- Druhy fulltextových indexů
  - CTXCAT
    - Vhodný pro menší dokumenty a jejich úryvky
    - Může být zkombinován s dalšími netextovými sloupci pro kombinované dotazování
    - Synchronizace indexu s daty se provádí automaticky se změnami v tabulce (obdoba WITH CHANGE_TRACKING AUTO z MS SQL)

Oracle fulltext

- Druhy fulltextových indexů
  - CTXRULE
    - Postaven na množině předdefinovaných dotazů
    - Slouží pro klasifikaci dokumentů do skupin podle toho, kterým dotazům vyhovuje

Oracle fulltext

- Uložení dokumentů
  - NORMAL_DATASTORE
    - Text je v jednom sloupci jednoho řádku
  - MULTI_COLUMN_DATASTORE
    - Text je ve více sloupcích jednoho řádku
  - URL_DATASTORE
    - Text je na Internetu, dostupný přes URL ve sloupci
  - …

Oracle fulltext

- Vytvoření indexu nad textovým sloupcem

CREATE INDEX *index_name*
ON *table_name*(column_name)
INDEXTYPE IS
ctxsys.{context|ctxcat|ctxrule}
[PARAMETERS
('*param_name param_value* …)];

Oracle fulltext

- Potřebné zadat
  - Jméno tabulky, jméno textového sloupce typu
    char, *nchar*, *varchar2*, *nvarchar2*, *clob*, *nclob*
  - Typ indexu

Oracle fulltext

- Z dalších možností:
  - Filter … formát vstupních dat
  - Lexer … členění textu na slova s ohledem na jazyk, diakritiku a její přepisy (ö, oe, o), …
  - Datastore
  - Stoplist,
  - …

Oracle fulltext

- Příklad vytvoření indexu:

```
CREATE INDEX myindex
ON doc(htmlfile)
INDEXTYPE IS ctxsys.context
PARAMETERS(
'datastore ctxsys.default_datastore
 filter ctxsys.null_filter
 section group ctxsys.html_section_group');
```

Dotazování se nad textovým indexem

- Spolu s novými typy indexů databáze implementují nové operátory pro porovnávání dotazu s textem
- Operátory vrací číslo – očekávanou míru shody obsahu textu s tazatelovými požadavky

Dotazování se nad textovým indexem

- Oracle používá numerickou funkci
  CONTAINS(
  ***sloupec*,**
  ***'dotaz'*[,**
  číslo_porovnání
  ])
- Pomocí SCORE(*číslo_porovnání*) lze v SELECT části zjistit hodnotu operátoru

Dotazování se nad textovým indexem

- Operátory pro dotazování
  - AND, & … 'mice & cats'
  - OR, | … 'mice | cats'
  - NOT, ~ … 'mice ~ cats'
  - NEAR, ; … proximitní dotazování
    'mice ; cats'
    NEAR((mice,cats),5)

Dotazování se nad textovým indexem

- Operátory pro dotazování
  - ABOUT (*téma*)
    dokumenty pojednávající o tématu
    'about(politics)'
  - WEIGHT, *
    vynásobí skóre výrazu danou konstantou
  - STEM, $
    nalezne termy se stejným slovním základem

Dotazování se nad textovým indexem

- Operátory pro dotazování
  - Na začátku / konci slova je možné uvést zástupné symboly _ a %
  - *dotaz* WITHIN PARAGRAPH
    dotaz WITHIN SENTENCE
    - Hledání v rámci odstavce, věty

Dotazování se nad textovým indexem

```
SELECT nazev, SCORE(1)+SCORE(2)
FROM dokument
WHERE
CONTAINS(abstrakt,'database &
search',1) > 0
AND
CONTAINS(
  text,
  '(object|relational) NEAR database',
  2) > 0
```

Dotazování se nad textovým indexem

- Využití tezauru
  - BT(*slovo*[,*n*]) … zahrnout i širší term pro *slovo* pokud je slovo homonymní, zahrnout všechny
  - NT(*slovo*[,*n*]) … zahrnout i užší termy pro *slovo*
  - PT(*slovo*) … nahradit preferovaným termem pro *slovo*
  - TT(*slovo*) … nahradit nejširším termem pro *slovo*
  - SYN(*slovo*) … zahrnout i synonyma pro *slovo*

Synonyma v Oracle

- Vytvoření
  CREATE
  [PUBLIC] SYNONYM
  jmeno
  FOR
  [schema.]stare_jme
  no;
- Zrušení
  DROP [PUBLIC]
  SYNONYM jmeno;

- Všude, kde lze použít původní název, lze použít nové jméno
- Lze vytvářet synonyma téměř ke všemu včetně synonym
- Veřejná synonyma vidí všichni bez prefixu vlastníka