

Információ-visszakeresés elmélete

Jegyzet

Ekart Csaba

2017/2018 ősz

Tartalomjegyzék

1. Az információ-visszakeresés alapjai és története	2
1.1. Az információ-visszakeresés fogalma	2
1.2. Web és az internet	2
1.2.1. A web története	2
1.2.2. A szemantikus világháló	3
1.2.3. Böngészés és keresés	3
1.3. Az információ visszakeresés alapfogalmai	3
1.4. Az információ visszakeresés kialakulása	4
1.5. Az információ visszakeresés kapcsolódó területei	4
1.6. Adattípusok	4
1.6.1. Strukturálatlan adat	4
1.6.2. Félig-strukturált adat	4
1.6.3. Strukturált adat	4
1.7. Adatbázis kezelő és információ-visszakereső rendszerek	4
1.8. Szövegbányászat és az információ-visszakeresés összehasonlítása	5
1.9. A nyelvtechnológia	5
1.10. Az információ-visszakeresés fogalmának kialakulása	5
1.11. Formális definíció	5
2. Az információ visszakereső modellek	5
2.1. Az információ-visszakereső modellek áttekintése	5
2.2. Klasszikus információ-visszakereső modellek	6
2.3. Az információ visszakereső modellek csoportosítása	6
2.4. Boole modell	7
2.4.1. Bevezetés	7
2.4.2. Formális leírása	7
2.5. Vektortér modellek	9
2.5.1. Formális leírás	9
2.5.2. Hatványtörvény	10
2.5.3. ZIPF-törvény	11
2.5.4. Indexelési technikák, hasonlósági mértékek	11
2.5.5. A visszakeresés lépései	12
2.5.6. Hasonlósági mértékek	13
2.5.7. A rangsortartás	13
2.5.8. Példa	14
2.6. Általános információ visszakereső rendszer architektúrája	14
2.6.1. Részfolyamatok	14
2.6.2. A információ-visszakereső rendszer elemei	15

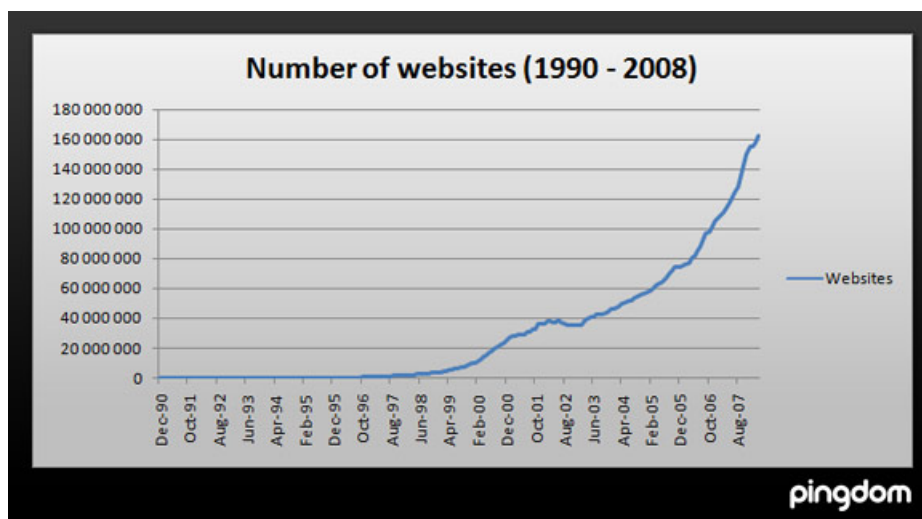
1. Az információ-visszakeresés alapjai és története

1.1. Az információ-visszakeresés fogalma

Az információ-visszakeresésen olyan strukturálatlan, többnyire szöveges anyagok (dokumentumok keresését) keresését jelentő tevékenységet értünk, amelynek célja a kereső információigényének kielégítése dokumentumok sokasága alapján. Nem összekeverendő az információ kinyeréssel, amely strukturálatlan szövegből strukturáltat állít elő.

1.2. Web és az internet

Az internet olyan globális számítógépes hálózatok hálózata, amely felhasználók milliárdjait kapcsolja össze. Ezzel szemben a világháló (World Wide Web (WWW, web)) az interneten működő, egymással hivatkozásokkal (hyperlinkekkel) összekötött weblapok és dokumentumok hálózata. A webet böngészőprogramokkal (browserekkel) tudjuk elérni. A weblapok a világ különböző helyein lévő számítógépeken vannak. Tehát az internet egy fizikai hálózat, a web pedig az interneten elérhető egyik szolgáltatás. Jelenleg kb. 1 milliárd weboldal található rajta, és 4 milliárd felhasználója van.



A web kihasználtságát mutatja, hogy manapság körülbelül 3-4 milliárd keresés történik a Google-on 24 óra alatt, és 170 milliárd e-mail küldés történik (nagyreszes SPAM).

1.2.1. A web története

A web történetének legfontosabb mérföldkövei:

- 1989: Javaslat egy rendszer hálózat kialakítására
- 1991: T. Berners-Lee elérhetővé teszi az első weboldalt, ami tartalmazza a világháló koncepcióját, és a böngészőkkel kapcsolatos alapszabványokat
- 1993: A világháló mindenki számára elérhetővé és ingyenesé válik.

A weboldalak számának gyarapodását egy korábban beszúrt grafikon remekül mutatja.

A webkülönböző verzióinak ismertetése e:

- **Web 1.0 - 1990-től 2001-ig**
 - A web hőskora
 - statikus weblapok
 - Papír világ digitális leképezése a cél

- Email
- Cégek, magánszemélyek, hírodalak
- Címszavak: You and I, Bring the web into our lives

- **Web 2.0 - 2001-től napjainkig**

- Dotcom válság után
- Felhasználók által generált tartalom, közösségi média
- Blogok
- Wikipedia
- Címszavak: Us, Bring our lives into the web

Jelenleg nem tudjuk mikor érkezik a Web 3.0, feltehetőleg az érzékelés és érzelmek, különböző szociális tartalmak, és a virtuális valóság fogják meghatározni.

1.2.2. A szemantikus világháló

A szemantikus világhálónak szokták nevezni azokat a célkitűzéseket, melyek egy olyan infrastruktúra létrehozását kívánják elhozni, mely lehetővé teszi a webes adatok integrálását és a közöttük lévő kapcsolatok definiálását, értelmezését, illetve az adatok értelmezését.

Ha a szemantikus világháló létrejönne akár fel is tehetnénk kérdéseket a webkeresőnek és az válaszolna (in progress... Lásd Siri, Google Assistant, Cortana, Alexa...) Jelenleg az internetes adatok nincsenek megfelelően kategorizálva ehhez. Azon honlapok, melyek saját adatbázisukból építkeznek, már konkrétan adatokkal tudnak szolgálni ilyen kérdések esetén (lásd: Wolfram Alpha)

1.2.3. Böngészés és keresés

Böngészésnek nevezzük egy adott webcím megadásán keresztül egy honlap felkeresését, majd onnan az oldalon található linkek segítségével más webhelyekre jutást.

A keresés során nem egy adott weboldalt keresünk fel, hanem egy keresőprogram segítségével keresünk számunkra hasznos weboldalakat. **A keresés típusai:**

- **Tematikus keresés:** különböző weblapok témájuk szerint témakörökbe, és alcsoportokba történő besorolása. (pl. Startlap)
- **Kulcsszavas keresés:** Egy keresőkérdést kell megadni, melynek eredményeként a kulcsszónak megfelelő weblapok listáját találjuk, amely az oldal címét, a szöveg előfordulását, egy kis részletet jelez a talált webhelyről. (pl. Google)

1.3. Az információ visszakeresés alapfogalmai

- **Tartalomjegyzék:** Az ókori görögök és rómaiak papirusztekercsekre írtak, jegyzeteket készítettek arról, hogy mely fontos rész hol található az adott műben. Valerius Soranus használt először tartalomjegyzéket.
- **Betűrend:** szavak ilyen típusú rendezését a görög tudósok alkalmazták először az elxandriai könyvtárban, hogy könnyebben eligazodjanak a művek között.
- **Hierarchia:** Egy írott dokumentumon könnyebb eligazodni, ha fejezetekre, alfejezetekre tagolódik. Római tudósok használták először ezt a módszert. Az írott szöveg szervezése az őse a fastruktúrának.
- **Index:** Az indexálás egy olyan eljárás, amely megadja
 - egy azonosítónak (szó, kifejezés, név tárgy, kód)
 - egy egységben (könyv, adatbázis) elfoglalt
 - pontos helyét (fejezet, oldal, sor)

1.4. Az információ visszakeresés kialakulása

Az információ visszakeresés alapötlete a hidegháboró idején vetődött fel az USA-ban, hogy az adatbázisokat hatékonyabban lehessen kezelni.

Miután az oroszok fellőtték az első máholdat, Amerika egyre több pénzt fektetett a különböző tudományok fejlesztésére.

A 90-es évekre már több információ visszakerső módszer volt ismert. Laboratóriumokban több kísérleti rendszert teszteltek, és sok kísérleti anyag gyűlt össze. Az évtized vége megjelentek a web-keresőmotorok.

1.5. Az információ visszakeresés kapcsolódó területei

Az információ visszakeresés rokonterületei:

- Adatbáziskezelés
- Webbányászat
- Adatbányász
- Nyelvtechnológia elmélete

1.6. Adattípusok

1.6.1. Strukturálatlan adat

A strukturálatlan adat elektronikus formában tárolt adat, amelyre nem illeszthető jól használható adatmodell. Az adatelemzés szempontjából ezek elemzése a legnehezebb. (pl. videó, audió, hosszú szöveges adatok (blog stb.))

1.6.2. Félig-strukturált adat

Átmenezik a strukturálatlan és a jól strukturált adat között. Tárolási formát tekintve tartalmaznak olyan formális elemeket, amelyek elkülönítik az egyes tartalmi részeket egymástól, de ezen tagolás még távol áll a strukturált formában megadott reprezentációtól (pl. táblázat). Példa félig strukturált adatra az email, amin a feladó, a címzett a tárgy mind strukturált formában van, viszont a tartalmi rész strukturálatlan marad.

1.6.3. Strukturált adat

Az információ elemi szintű adatokra bomlik, s ezen elemi adatok kapcsolat rendszere modellek által definiált. A strukturált formába rögzített adatokon leggyakrabban táblázatokban tárolt adatokat értünk, ahol az egyes oszlopok az objektumokat leíró tulajdonságokat tartalmazzák, egy-egy sor pedig egy-egy objektumnak feleltethet meg.

1.7. Adatbázis kezelő és információ-visszakereső rendszerek

- **Az adatbázis kezelő:**
Strukturált adatokon dolgozik, ahol jól definiált a struktúra, a szemantika, valamint a lekérdezés is strukturált formában van megadva.
- **Információ visszakereső rendszer:**
Strukturálatlan adatokon dolgozik, ahol természetes nyelvű szövegek vannak, és a szemantika is többértelmű, valamint a keresőkérdés természetes nyelvű szöveg.

1.8. Szövegbányászat és az információs-visszakeresés összehasonlítása

- **Szövegbányászat:**

Új, a felhasználó számára eddig ismeretlen összefüggések, ismeretek kinyerése. Olyan tudásra kívánunk szert tenni, ami nincs közvetlen rendelkezésre álló dokumentum állományában, csak elrejtve a tartalmában.

- **Információ visszakereső rendszer:**

A felhasználónak konkrét információ igénye van, és a dokumentumokban fellelhető információk alapján próbáljuk meg kielégíteni azt.

A kettő tudománynak tehát közös tulajdonsága, hogy mindkettő strukturálatlan szöveges állományban keres.

1.9. A nyelvtechnológia

A számítógépes nyelvészet tudománya. Természetes nyelvű szövegek számítógépes feldolgozásával foglalkozik.

A gép a nyelvet szöveggént, sőt betűk sorozataként érzékeli, számára ezek számkódok sorozata. Nem csak tárolnia és megjelenítenie kell a szöveget, de ismernie a nyelvi szerkezeteket is. A természetes nyelvekben szízbálszerűségek vannak, és ezeket meg kell találnia. A legalapvetőbb leírási szintje a számítógépes morfológia és szintaxis, melynek bonyolultsága nyelvfüggő.

1.10. Az információ-visszakeresés fogalmának kialakulása

1.11. Formális definíció

Legyen IR az alábbi:

$$IR = (U; IN; Q; O) \rightarrow R$$

ahol

- $U = felhasználó$
- $IN = információigény$
- $Q = keresőkérdés$
- $O = keresendőobjektumokhalmaza$
- $R = aQkérdésreválaszkéntvisszaadottobjektumokhalmaza$

Az információigény (IN) alatt az alábbi

$$IN = (Q, I)$$

kifejezést értjük, ahol az I a felhasználó-specifikus információtöbbséget jelenti, mely nem foglalmazódik meg a keresőkérdésben.

2. Az információ visszakereső modellek

2.1. Az információ-visszakereső modellek áttekintése

Az információ-visszakeresés alapelemei a

- dokumentum (document)
- kérdés (query)
- relevancia (relevance)
- visszakeresés (retrieval)

Attól függően, hogy a fentieket hogyan modellezzük, többféle információ-visszakereső modellt különböztetünk meg.

2.2. Klasszikus információ-visszakereső modellek

- Az első hagyományos modellek ilyenek
- Matematikai módszereken alapszik
- Dokumentum és kérdés "távolságának" mérésén alapszik
- Könnyű implementálni, ezért a kereskedelmi keresők gyakran használják

A legfontosabb klasszikus információ-visszakereső modellek:

- Boole modell (matematikai logika)
- Vektortér modell (lineáris algebra)
- Valószínűségi modell (valószínűségszámítás)

A nem klasszikus, alternatív modellek az internetes, hálózati környezet tapasztalatait is felhasználják, tehát nem csak magukat a dokumentumokat vizsgálja, hanem az azok egymáshoz való viszonyát is. Ilyenek például a

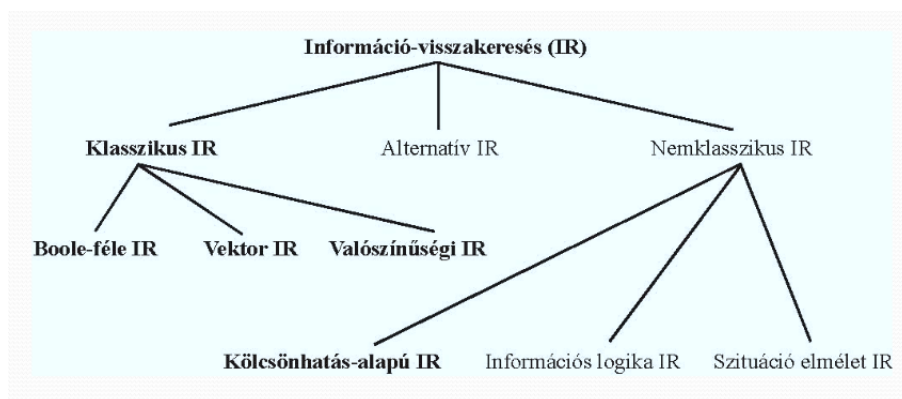
- Információs logika alapú
- Szituációelmélet alapú
- Kölcsönhatás alapú
 - nem egymástól elszigetelt egységet képeznek, hanem egy összekötött hálózatot.

modellek. Egyéb alternatív modellek a

- KLASZTER-alapú
- Fuzzy-alapú
- Genetikus algoritmus
- Tudásbázis

alapú modellek.

2.3. Az információ visszakereső modellek csoportosítása



2.4. Boole modell

2.4.1. Bevezetés

- A legelső modell
- mely széleskörben elterjedt
- Kereskedelmi keresők alapjául szolgál
- A Boole logikára és a klasszikus halmazelméletre épül
- A Q kérdést és a D dokumentumokat is szavak halmazaként kezeli.

A Boole modellben a visszakeresés azon alaul, hogy a dokumentum tartalmazza e, vagy nem a keresőkérdésben megadott kifejezéseket.

2.4.2. Formális leírása

A Boole modell (Boolean Information retrieval Model) formálisan: Adottak:

- $D = \{D_1, \dots, D_m\}$ dokumentumok és
- $T = \{t_1, \dots, t_n\}$ indexkifejezések, amelyek leírják a dokumentumokat, illetve egy
- Q kereső kérdés (egy Boole kifejezés)

A dokumentumok tulajdonképpen indexkifejezések halmazának tekinthetők.

A valóságban a dokumentumok és a reprezentációjuk két külön entitás, matematikailag viszont ekvivalensnek tekinthetők, hiszen a dokumentumokat a rájuk jellemző indexkifejezésekkel reprezentálhatjuk, valamint a visszakeresés során is a dokumentumok reprezentációját használjuk nem magukat a dokumentumokat.

Az algoritmus

1. lépés:

Meghatározzuk a dokumentok azon S_i halmazát, amely tartalmazza vagy nem a t_i kifejezést:

- t_i kifejezésre: $S_i = \{D | t_i \in D\}$
- $\neg t_i$ kifejezésre: $S_i = \{D | t_i \notin D\}$

2. lépés:

A Q kérdésre válaszként visszaadott dokumentumokat a logikai műveleteknek megfelelő halmazműveletek adják meg:

- $\cap$ (metszet) megfelel a $\vee$ (logikai és) kapcsolatnak
- $\cup$ (unió) megfelel a $\wedge$ (logikai vagy) kapcsolatnak

Példa: $Q = t_1 \vee (t_2 \wedge t_3)$

- S_1 eredményhalmaz t_1 indexkifejezésre
- S_2 eredményhalmaz t_2 indexkifejezésre
- S_3 eredményhalmaz t_3 indexkifejezésre

A Q keresőkérdésre visszkapott eredményhalmaz:

$$S_1 \cup (S_2 \cap S_3)$$

Példa a Boole modell használatára

- Legyen a valós dokumentumok halmaza O az alábbi:

$$O = \{O_1, O_2, O_3\}$$

- Legyen a T indexkifejezések halmaza az alábbi $T = \{t_1, \dots, t_6\}$, ahol

- $t_1 = \text{virág}$
- $t_2 = \text{tél}$
- $t_3 = \text{hó}$
- $t_4 = \text{fenyő}$
- $t_5 = \text{bunda}$
- $t_6 = \text{varjú}$

O_1	O_2	O_3
Még nyílnak a völgyben a kerti virágok , Még zöldell a nyárfa az ablak előtt, De látod amottan a téli világot? Már hó takarja el a bérci tetőt.	Fenyő ága Hósubában , Mire vársz a Hófúvásban ? Hideg az a Kristály bunda , Gyere haza Kis házunkba.	Fekete Pont Fehér Ágon: Varjú károg: Fázom Fázom.

- **Feladat:** Adja meg a $T = \{t_1, \dots, t_6\}$ indexkifejezésekkel az $O = \{O_1, O_2, O_3\}$ objektumokat reprezentáló $D = \{D_1, D_2, D_3\}$ dokumentumhalmazt!

- **Megoldás:**

- $D_1 = \{\text{virág}, \text{tél}, \text{hó}\} = \{t_1, t_2, t_3\}$
- $D_2 = \{\text{hó}, \text{fenyő}, \text{bunda}\} = \{t_3, t_4, t_5\}$
- $D_3 = \{\text{varjú}\} = \{t_6\}$

- **Feladat:** Legyen a Q kereső kérdés az alábbi: $\text{hó} \wedge \text{fenyő}$, azaz

$$Q = t_3 \wedge t_4$$

- **Megoldás:** Ekkor az S_3 és S_4 a következő:

- $S_3 = \{D_1, D_2\}$ azon dokumentumok halmaza, amelyek tartalmazzák a $t_3 = \text{hó}$ indexkifejezést, míg
- $S_4 = \{D_2\}$ azon dokumentumok halmaza, amelyek tartalmazzák a $t_4 = \text{fenyő}$ indexkifejezést.

A Q keresőkérdésre bálaszként adott dokumentumhalmaz, tehát a következő:

$$S_3 \cap S_4 = \{D_2\}$$

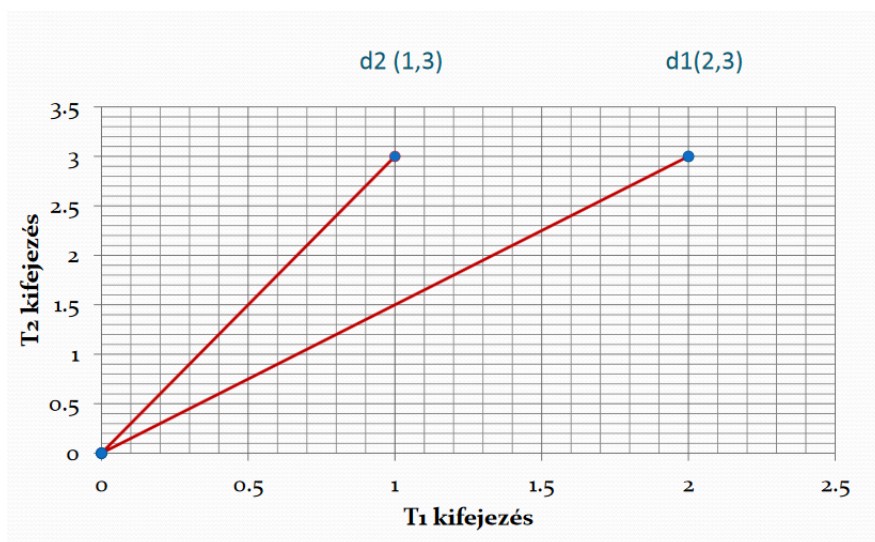
Tehát a Q keresőkérdésre adott válasz az O_2 objektum lesz.

2.5. Vektortér modellek

A vektortér modell egy fontos, jól érthető, és széles körben kutatt és használt klasszikus modell, amelyet szöveges objektumok feldolgozására és információ-visszakeresésre már régóta használnak.

Ezt a vektortér modellnek nevezzük, mert minden dokumentum és kérdés a tér egy pontjába van leképezve, mely tér alapját a dokumentumokban található kifejezések adják.

- A tér matematikai modellje egy orthonormált euklideszi tér, amelyben a tengelyek páronként egymásra merőlegesek.
- A tér dimenzióit az indexkifejezések adják.
- A visszakeresés azon alapul, hogy a kérdés-vektor és a dokumentum-vektor mennyire van "közel" egymáshoz.



2.5.1. Formális leírás

Legyen

- D egy véges halmaz, melynek elemei a dokumentumok:

$$D = \{D_1, \dots, D_m\}$$

- T egy véges halmaz, melynek elemei az indexkifejezések:

$$T = \{t_1, \dots, t_n\}$$

- Minden D_j dokumentumhoz hozzárendelünk egy n hosszú v_j súlyvektort. A vektor elemeit súlyoknak nevezzük. A súllyal azt fejezzük ki, hogy a t_i kifejezés milyen mértékben tükrözi a D_j tartalmát.

A súlyvektorok segítségével megadható az ún. TxD (term-by-document) kifejezés-dokumentum mátrix, melyre

- m oszlopa van (dokumentumok száma)
- n sora van (indexkifejezések száma)
- amelynek elemei a súlyok

$$TD = (w_{ij}) = \begin{bmatrix} w_{11} & w_{12} & w_{13} & \dots & w_{1m} \\ w_{21} & w_{22} & w_{23} & \dots & w_{2m} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ w_{n1} & w_{n2} & w_{n3} & \dots & w_{nm} \end{bmatrix}$$

A kifejezések kiválasztása és a súlyok meghatározása nehéz elméleti és gyakorlati probléma. Ennek számos lehetséges megoldása van:

- A legnyilvánvalóbb az, hogy az index-kifejezéseket magukban a dokumentumokban keressük.
- Feltételezzük, hogy a szavak előfordulási gyakorisága a dokumentumokban jelentőséggel bír, és ezért azonosítóként használhatók.

2.5.2. Hatványtörvény

Néhány alapfogalom tisztázása a törvény kimondása előtt.

- **Fokszám**nak nevezzük egy hálózat egy elemének a hálózaton belüli kapcsolatainak számát.
- **Fokszámeloszlás** alatt a hálózat összes adott fokszámú elemének számát tüntetjük fel a fokszám függvényében.

Véletlen hálózat fokszáma Poisson-eloszlást követ (haranggörbe). A

- A legtöbb csomópontnak azonos számú kapcsolata van
- nem létezik kiemelkedően sok kapcsolatú csomópont.

Skálafüggetlen hálózat hatványfüggvényű fokszámeloszlású

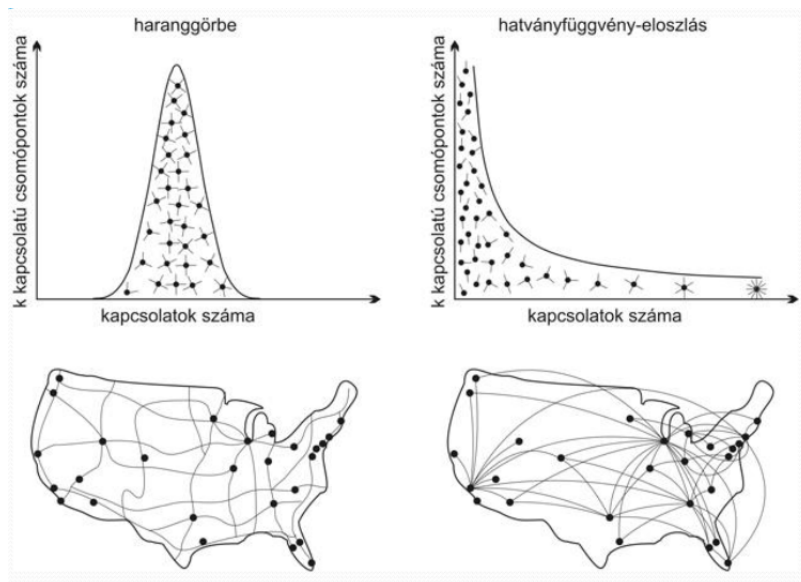
- a legtöbb csomópontnak csupán kevés kapcsolata van,
- amelyeket néhány nagymértékben összekapcsolt középpont tart össze. (Barabási, 2003 után)

Ha egy esemény változása valamely jellemzőjére hatványával arányos, akkor azt mondjuk, hogy a hatványtörvény szerint viselkedik.

A hatványtörvénnyel leírt hálózatok esetében az elemek fokszámeloszlása szabályszerű:

- van kisszámú elem, aminek nagyon sok kapcsolata van (pl. 2 db 100 kapcsolattal rendelkező) kevés amire sok link mutat
- majd a kapcsolatok számának csökkenésével növekszik az adott kapcsolattal rendelkező elemek száma amelyekre kevés link mutat.

Azon hálózatokat, amelyeknél az elemek fokszám-eloszlása a hatványtörvényt követi skálafüggetlen hálózatoknak nevezzük.



Példa 203 millió weboldal vizsgálata (bejövő linkek alapján):

- Oldalak 90%-a **10 vagy kevesebb** beérkező linkkel rendelkezik.
- Néhányat (3) közel **1 millió** másik oldalon hivatkoznak.

2.5.3. ZIPF-törvény

Angol nyelvű szövegekben (korpuszokban) a szavak előfordulása gyakorisága a hatványtörvényt követi:

$$f(r) = Cr^{-\alpha}$$

- Ahol C korpusz függő konstans,
- r a szavak rangsora
- α a hatványfüggvény kitevője.

Tehát a Zipf-törvény azt állítja, hogy egy természetes nyelv egyes részeiben egy szó előfordulási gyakorisága fordítottan arányos a gyakorisági táblában lévő rangjával. Tehát a leggyakoribb szó kb. 2x gyakoribb, mint a második leggyakoribb szó és háromszor gyakoribb, mint a harmadik helyen lévő stb.

Példa Az úgynevezett Brown-gyűjteményt, mely kb. 500 angol szöveget tartalmaz megfigyelték, hogy a leggyakoribb szó a "the" volt, mely az összes előforduló szó 7%-át tette ki, az "and", pedig a második, mely pedig 3,5%-on áll.

Egy következmény A törvény érdekes következménye, hogy egy korpuszból csak a leggyakoribb szavakat tartjuk meg, a többit töröljük a korpusz nagy része továbbra is megmarad.

2.5.4. Indexelési technikák, hasonlósági mértékek

Az automatikus indexkifejezés-kinyerés lépései A kifejezések és azok jelentőségének meghatározására a következő automatikus módszer használható:

1. Lexikai egységek azonosítása. Egy számítógépes program kell szavak felismerésére.
2. Stoplista alkalmazása (azokat a szavakat tartalmazza, amelyek általában nem hordoznak jelentést egy dokumentumban (pl. a, az, egy ...). Azokat a szavakat, amelyeket a stoplista tartalmaz, kihagyjuk a további vizsgálatkor.

3. Szótövesítő (stemming) algoritmus alkalmazása. Ez az algoritmus minden szót redukál vagy áttanszformál a nyelvi szótőre.
4. Kiszámítjuk minden D_j dokumentumra a t_i kifejezés előfordulásának számát: f_{ij}
5. Kiszámítjuk a t_i kifejezés összes előfordulását: t_{fi}

$$t_{fi} = \sum_{j=1}^m f_{ij}$$

6. t_{fi} szerint sorba rendezzük a kifejezéseket
7. a nagyon magas értékűeket, és a nagyon alacsony előfordulásúakat kirekesztjük
8. Az így megmaradó kifejezések az azonosítók vagy indexkifejezések.
9. Az indexkifejezések felhasználásával számítjuk minden D_j dokumentumra a W_{ij} súlyokat. Súlyszámuk segítségével fejezzük ki, hogy egy kifejezés milyen mértékben tükrözi egy dokumentum tartalmát. A súlyszámok meghatározására számos módszer használható.

- **Bináris:**

$$w_{ij} = \begin{cases} 1, & \text{ha a } t_i \text{ kifejezés szerepel a } D_j \text{ dokumentumban} \\ 0, & \text{egyébként} \end{cases}$$

- **Gyakoriság szerinti súlyozás (TF: Termfrequency):**

A súlyfüggvény megegyezik a kifejezések előfordulási gyakoriságával.

$$w_{ij} = f_{ij}$$

- **Normalizált gyakoriság szerinti súlyozások:**

- Maxnormált

$$w_{ij} = \frac{f_{ij}}{\max_k f_{kj}}$$

- Hossznormált

$$w_{ij} = \frac{f_{ij}}{\sqrt{\sum_{k=1}^n f_{kj}^2}}$$

- Logaritmusos (TF-IDF)

$$w_{ij} = f_{ij} \times \log \frac{m}{F_i}$$

ahol m a dokumentumok száma, F_i pedig azon dokumentumok száma, amelyekben előfordult a t_i indexkifejezés.

2.5.5. A visszakeresés lépései

A visszakeresés azon alapul, hogy a kérdés-vektor és a dokumentum-vektor mennyire van közel egymáshoz. A felhasználó által feltett Q_k keresőkérdéshez is megadható egy v_k vektor, amelynek elemeit ugyanolyan súlyszámítási séma alapján adjuk meg, mint a dokumentumokét:

$$v_k = (w_{ik}) = (w_{1k}, \dots, w_{nk})$$

ahol a w_{ik} a t_i indexkifejezés Q_k keresőkérdésre vonatkozó súlyszáma.

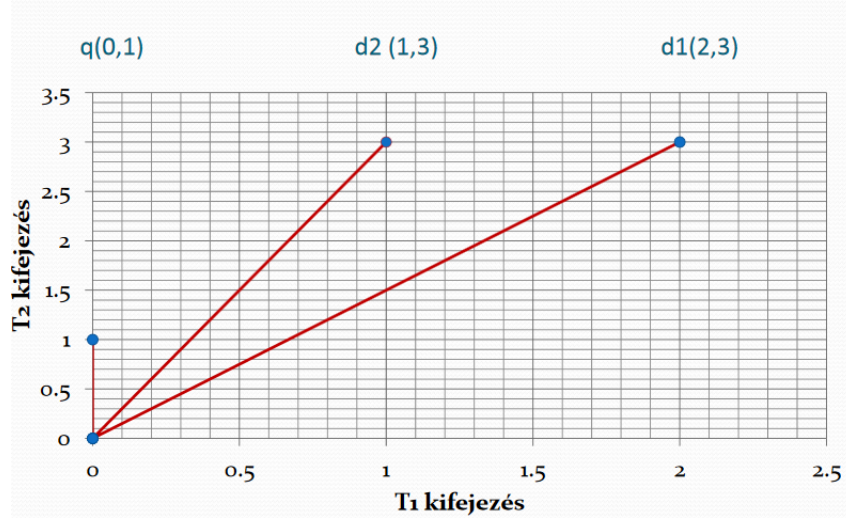
A visszakeresés egy hasonlóság meghatározásán alapul.

- A dokumentumoknak megfelelő v_i vektorokat összevetjük a kérdésvektorral v_k . Ennek eredménye s_{ik}
- Ha ez a mérési eredmény egy küszöbértéknél nagyobb akkor a v_i vektorral azonosított dokumentum válasz a kérdésre:

$$S_{ik} = s(v_i, v_k) > K$$

Tehát a visszakeresés lépései a következők:

1. A Q_k keresőkérdés megadása.
2. A t_i indexkifejezések w_{ik} súlyának meghatározása a Q_k kérdésben
3. Minden D_j dokumentumra a hasonlósági mérték értékeinek kiszámítása
4. A találati lista megadása a hasonlósági érték szerint csökkenő sorrendben.



2.5.6. Hasonlósági mértékek

- Pont vagy skalárszorzat

$$s_{jk} = (v_j, v_k) = \sum_{i=1}^n w_{ij} w_{ik}$$

- Koszinusz mérték

$$s_{jk} = c_{jk} = \frac{(v_j, v_k)}{\|v_j\| \cdot \|v_k\|} = \frac{\sum_{i=1}^n w_{ij} w_{ik}}{\sqrt{\sum_{i=1}^n w_{ij}^2 \sum_{i=1}^n w_{ik}^2}}$$

- Dice együttható

$$s_{jk} = d_{jk} = \frac{2 \cdot (v_j, v_k)}{\sum_{i=1}^n (w_{ij} + w_{ik})} = \frac{2 \cdot \sum_{i=1}^n w_{ij} w_{ik}}{\sum_{i=1}^n (w_{ij} + w_{ik})}$$

- Jaccard együttható

$$s_{jk} = J_{jk} = \frac{(v_j, v_k)}{\sum_{i=1}^n \frac{w_{ij} + w_{ik}}{2^{w_{ij} w_{ik}}}} = \frac{\sum_{i=1}^n w_{ij} w_{ik}}{\sum_{i=1}^n \frac{w_{ij} + w_{ik}}{2^{w_{ij} w_{ik}}}}$$

2.5.7. A rangsortartás

Adott

- Két tetszőleges dokumentum: D_1, D_2
- Két hasonlósági mérték: σ_1, σ_2

Ha a két dokumentum sorrendje megegyezik mindkét hasonlósági mértékkel számolva bármely tetszőleges Q keresőkérdésre, azaz

$$\sigma_1(w_1, q) \leq \sigma_1(w_2, q) \leftrightarrow \sigma_2(w_1, q) \leq \sigma_2(w_2, q)$$

A rangsortartás esetén tehát:

- a két hasonlósági mérték egymással ekvivalens, és helyettesíthető
- mivel ugyanazokat a találatokat adja vissza ugyanabban a sorrendben.

A vektortér modellben általában a hasonlósági mértékek egymással nem helyettesíthetők, tehát nem rang-sortartók.

2.5.8. Példa

- Adott könyvcímek egy kis gyűjteménye

Terms		Documents		Query
T ₁	Baby	D ₁	Infant and Toddler First Aid	
T ₂	Child	D ₂	Babies and Children's Room (For your Home)	Child
T ₃	Guide	D ₃	Child Safety at Home	
T ₄	Health	D ₄	Your Baby's Health and Safety: From Infant to Toddler	
T ₅	Home	D ₅	Baby Proofing Basics	Home
T ₆	Infant	D ₆	Your Guide to Easy Rust Proofing	Infant
T ₇	Proofing	D ₇	Babies Collectors Guide	Proofing
T ₈	Safety			Safety
T ₉	Toddler			

Table 3.1. Collection of book titles with the index terms

- $m = 7$ dokumentum azaz $D = \{D_1, \dots, D_7\}$
- $n = 9$ indexkifejezés (a mátrix 9 sora van), azaz $T = \{t_1, \dots, t_9\}$
- A doksik:
 - $D_1 = \{t_6, t_9\}$
 - $D_2 = \{t_1, t_2, t_5\}$
 - $D_3 = \{t_2, t_5, t_8\}$
 - $D_4 = \{t_1, t_4, t_6, t_8, t_9\}$
 - $D_5 = \{t_1, t_7\}$
 - $D_6 = \{t_3, t_7\}$
 - $D_7 = \{t_1, t_2\}$

Megoldás: tfn súlyszámítási sémával a TD mátrix:

$$D = \begin{bmatrix} 0 & 0.5774 & 0 & 0.4472 & 0.7071 & 0 & 0.7071 \\ 0 & 0.5774 & 0.5774 & 0 & 0 & 0 & 0.7071 \\ 0 & 0 & 0 & 0 & 0 & 0.7071 & 0 \\ 0 & 0 & 0 & 0.4472 & 0 & 0 & 0 \\ 0 & 0.5774 & 0.5774 & 0 & 0 & 0 & 0 \\ 0.7071 & 0 & 0 & 0.4472 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.7071 & 0.7071 & 0 \\ 0 & 0 & 0.5774 & 0.4472 & 0 & 0 & 0 \\ 0.7071 & 0 & 0 & 0.4472 & 0 & 0 & 0 \end{bmatrix}$$

2.6. Általános információ visszakereső rendszer architektúrája

2.6.1. Részfolyamatok

Technikai olfáról megközelítve az információ visszakereső rendszer az alábbi elemekből áll össze:

- egy objektumbázis (pl. weboldalak)

- egy keresőfelület

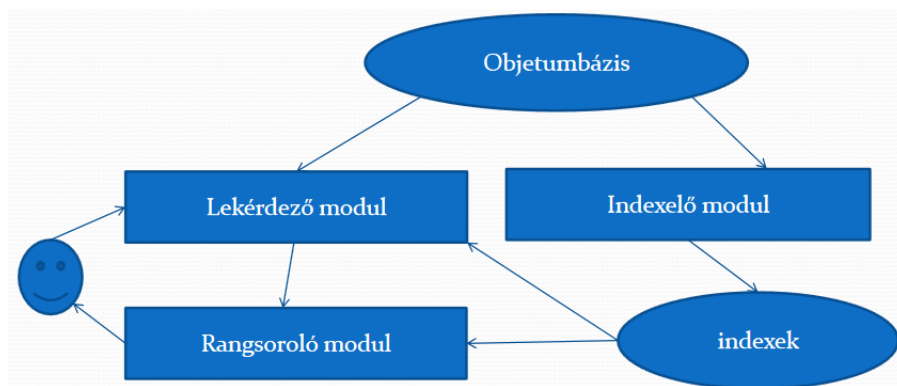
A releváns információ hatékony visszakeresését a felhasználói munka és a dokumentumok logikai nézete egyszerre befolyásolják.

Tehát az információ visszakeresési folyamat részfolyamatai:

1. **Objektumbázis definiálás:** dokumentumok, szövegfeldolgozó műveletek, szövegmodell megadása.
2. **Szövegfeldolgozó műveletek alkalmazása:** A műveletek átalakítják a dokumentumokat, és elkészítik azok logikai nézetét.
3. **Szöveg indexelése** Az indexelő modul végzik. Az index kritikus paraméter, gyors keresést tesz lehetővé. Többféle fájlstruktúra használható tárolására (pl. inverz fájl)
4. **Lekérdezés:** a lépés során az indexek és a keresőkérdés kifejezéseinek összehasonlítása történik.
5. **Rangsorolás:** A kapott eredmények alapján kiszámítjuk a hasonlósági értékeket.
6. **Találati lista visszaadása**

2.6.2. A információ-visszakereső rendszer elemei

- **Objektumbázis:** tárolja a dokumentumok halmazát.
- **Indexelő modul:** Az objektumbázisban lévő dokumentumokból az indexelő modul készíti az indexet invert fájlstruktúrában. Ezeket a struktúrákat használja majd a lekérdező modul, hogy megtalálja a felhasználó által feltett keresőkérdésre a válasz dokumentumokat.
- **Lekérdező modul:** A lekérdező modul olvassa be a keresőkérdést, és alakítja a felhasználható formára. A lekérdező modul az indexeket használja a felhasználói kérdésre megfelelő válaszok megtalálására. Ezután a talált dokumentumokat átadja a rangsoroló modulnak.
- **Rangsoroló modul:** A rangsoroló modul kiszámítja a lekérdező modul által talált dokumentumokra a hasonlósági értékeket, majd ezek alapján csökkenő sorba teszi a dokumentumokat.



Objektumbázis

Hivatkozások