Masovno otkrivanje dokumenata sličnih duplikata

Verzioniranje:

- Različite verzije jednog dokumenta
  - Revizije, različiti formati ...

Zrcaljenje

- Objavljeno na više mjesta

Plagiranje

- Identična ili obrađena kopija

Štetni sadržaj

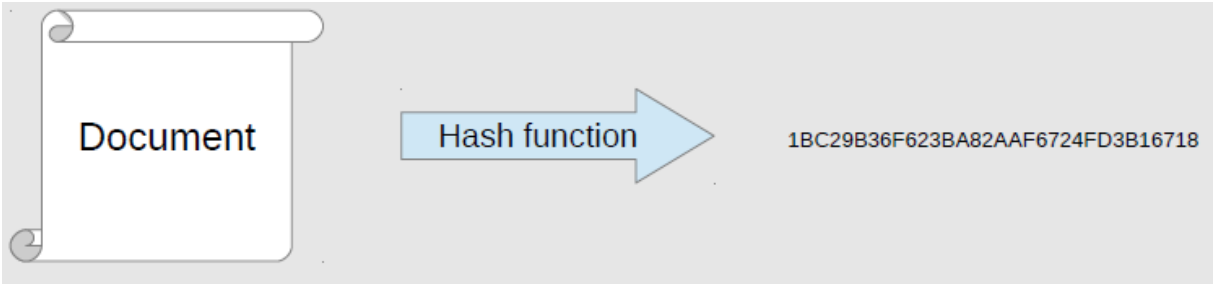
- Virusi, spam, ...

Skalabilno rješenje:

- Dokumenti i repozitoriji dokumenata su ogromni
  - Internet
  - Domena specifičnih tekstova
  - Logovi
  - ...

CheckSumming

- Kriptografske hash funkcije
- MD5, SHA1, SHA2, ...



Hvata i najmanje uređivanje

- Dobro za detekciju identičnih kopija
- Nije dobro za detekciju sličnih duplikata

DETEKCIJA SLIČNIH DUPLIKATA

Dvije metode:

- Fingerprinting (identifikacija)
  - Hashiranje dokumenta
- Rangiranje
  - Tehnike pronalaženja informacija

FINGERPRINTING

Sličnost čuvajući raspršivanje

- X, skup ulaza
- d_x, funkcija udaljenosti nad X
  - x1, x2 elements of X
- sličnost čuvajući hash funkciju
  - h: X → Y
  - |Y| < |X|
- d_y, funkcija udaljenosti nad Y
- slični ulazi imaju slične hash vrijednosti

$$\text{if } d_x(x1, x2) < \varepsilon_x, \text{onda}$$

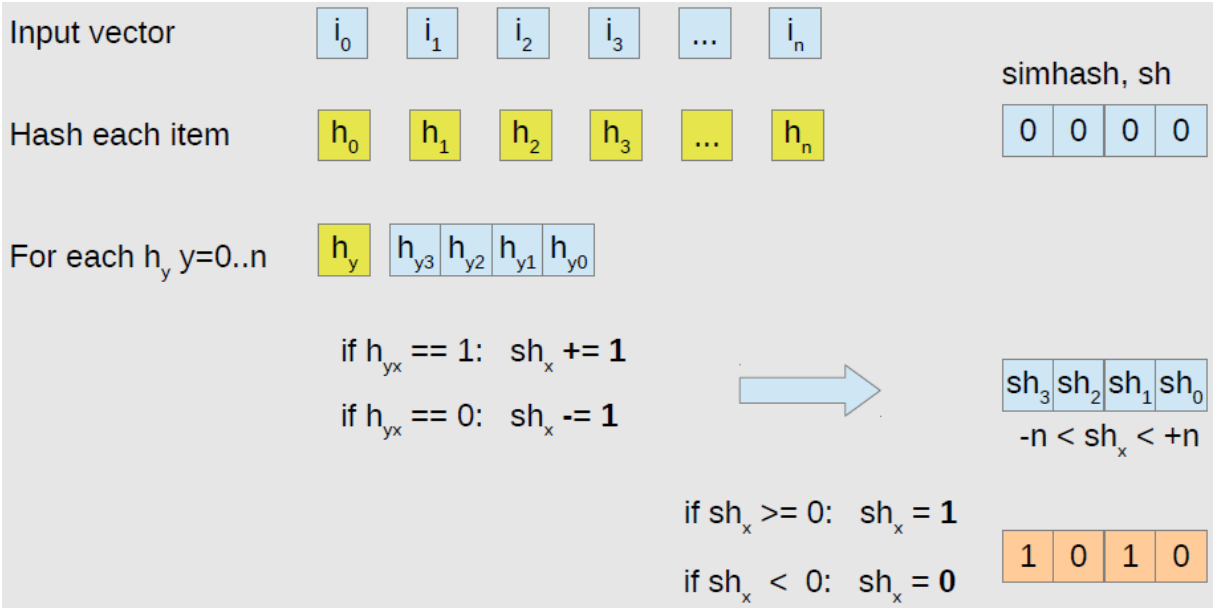
$$d_y(h(x1), h(x2)) < \varepsilon_y$$

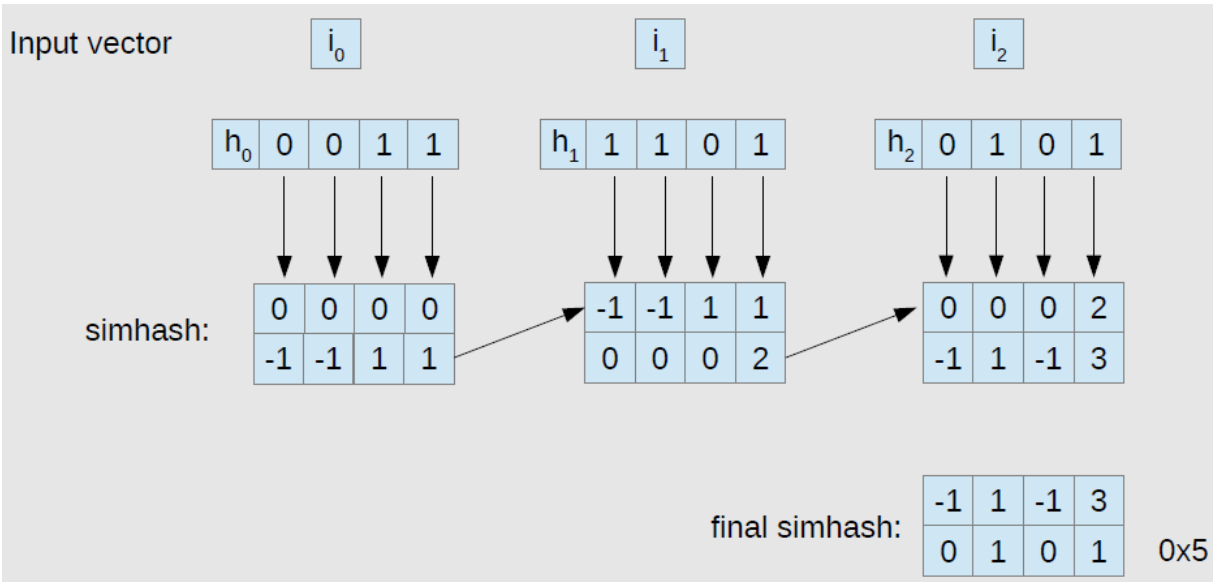
- primjer:
  - h(„text1“) = 0xaaaf
  - h(„text2“) = 0xaaae

Simhash algoritam

- Tehnika identifikacije
  - Identifikacija sličnih duplikata razlikuje se u malom broj pozicija bit-ova (hamingova udaljenost)
- Smanjenje dimanzionalnosti
  - Mape visoko-dimenzionalnih vektora za male fingerprintove (fbits)
- Veličina fingerprinta (fsize)
  - f je malen i proizvoljan

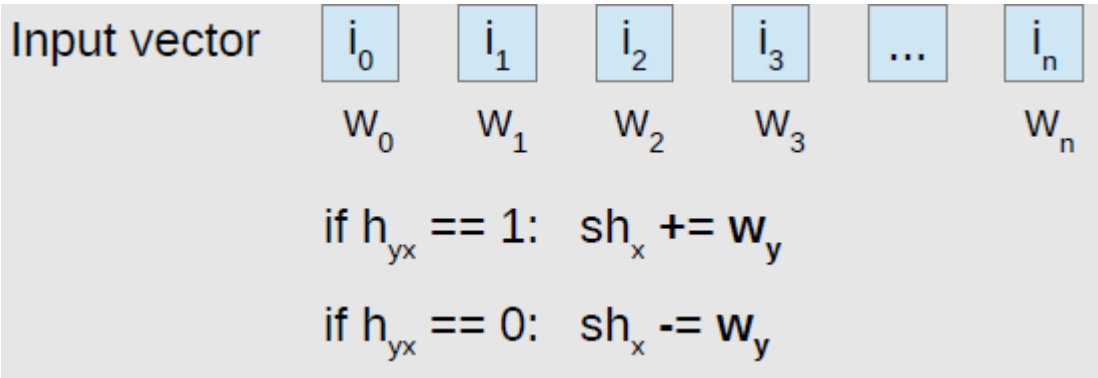
Računanje simhash-a za f=4





Ponderirana Simhash računanja

- Dodjela faktora težine na svaku novu značku



Izbor hash funkcije h

- Jedinstvena distribucija
- Brzina
- Kandidati: kriptografske hash funkcije
  - MD5(128bit), SHA-1(256bit)
- Vlastite hash funkcije sa duljinom varijable

Ulazni vektor

- Fokus na sirovom tekstu dokumanata
- Konvertira dokument u oblikovni vektor
- Prikupljanje podataka
  - Tokenizacija (shingling)
    - Unigrami, 2-grami, 3-grami
  - Proizlazi
  - Uklanjanje točke
  - Detekcija fraza
  - ...

“lorem ipsum dolor sit amet”

- “lorem”, “ipsum”, “dolor”, “sit”, “amet”
- “lorem ipsum”, “ipsum dolor”, “dolor sit”, “sit amet”
- “lor”, “ore”, “rem”, “em ”, “m i”, ...

- Shingling
    - Hash k-grama
    - k-grami
        - Znakovi, riječi, rečenice
    - k = ?
        - malen k: tazličiti dokumenti se pojavljuju slično
        - velik k: slični dokumenti izgledaju različito
    - oblikovni vektor iz IR izlaza
        - Ponderirana sa IDF
        - inverzna učestalost dokumenta
        - može se promijeniti kada se promijene kolekcije
    - Informacije povezivanja
        - Struktura poveznica (slične stranice imaju zajedničke poveznice)
    - Sidreni tekst
        - Slični dokumenti bi trebali imati sličan sidreni tekst (poveznice)

BRZI UPITI

- F – kolekcija f-bit fingerprintova
- Q – upit
    - Jedinstveni ii skup fingerprintova
- Zadatak
    - Identificirati kad god se Q razlikuje od ijednog fingerprinta F u najviše k bitova
- Google numbers
- 8B 64-bitni fingerprintovi = 64Gb
- Online upiti
    - Q = jedan fingerprint
    - Ograničenja: nekoliko milisekundi
- Skupni upiti
    - Q = skup fingerprintova
        - e.g |Q| = 1M
    - Ograničenja: ~100sekundi
        - 1B upita po danu

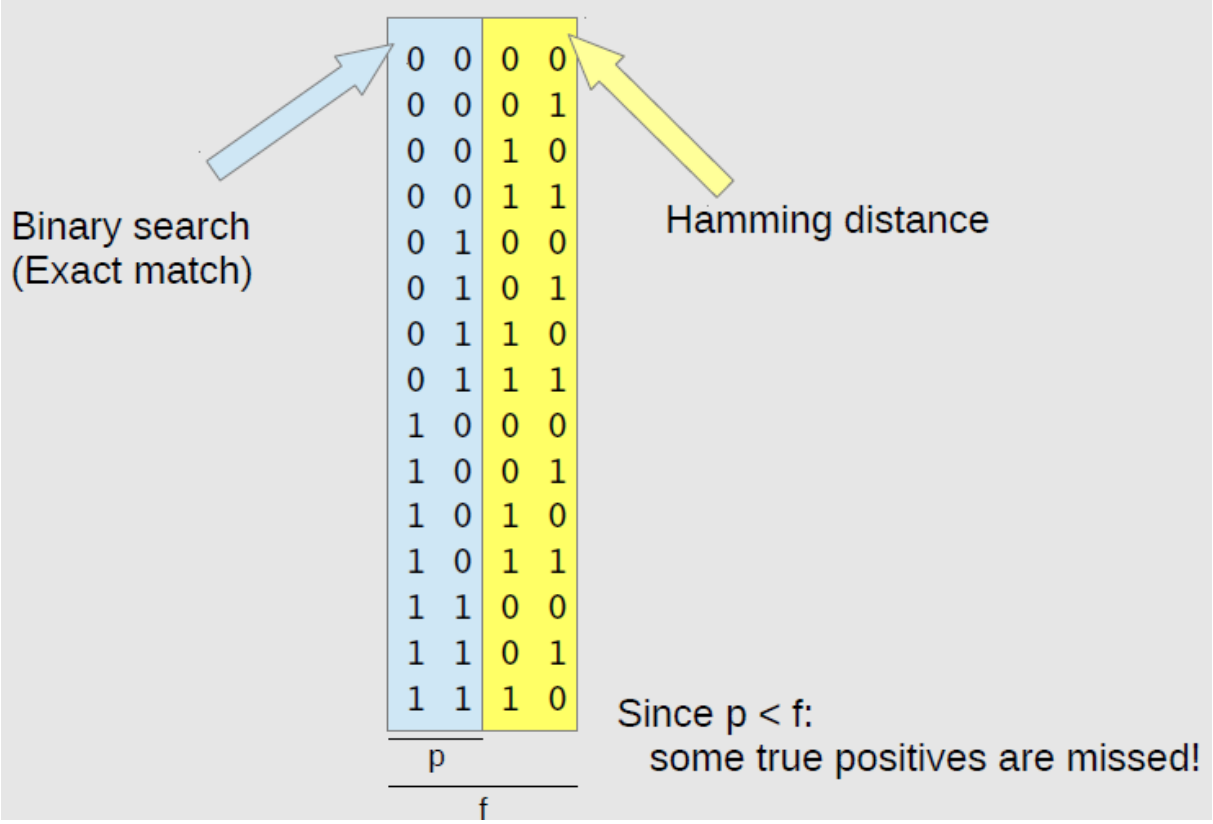
Prvi pristup:

- Napravi sortiranu tablu F-ova
- Napravi listu Q' sa svim fingerprintovima čija je hamingova udaljenost od Q najviše k

$$|Q| = \binom{64}{3} = 41664$$

Drugi pristup

- Napravi sortiranu tablicu F-ova
- Nađi skup fingerprintova (F') koji imaju jednako najviše značajnih djelova (p bitova)
    - Sortirana tablica – binarna pretraga 0(p).
- Provjeri hamingovu udaljenost za svaki fingerprint iz F'
- Ovaj pristup će locirati sve fingerprintove u F koji se razlikuju u najviše k bitova
    - Ograničeno na najmanje značajnih f-p bitova!



- Zamijenit žutu i plavu, + sort

Rješenje:

- Izgraditi dodatne tablice
    - Svaka sa različitom permutacijom bitova
    - Svaka tablica ima različit skup značajnih bitova
- Algoritam za brze (online) upite
    - Napravi t sortiranih tablica od fingerprintova: T1, T2, ..., Tt
    - Svaka tablica Ti sadrži
        - Pi – broj značajnih bitova
        - Πi – random permutacija
    - Svaki fingerprint u Ti je permutran sa permutacijom Πi

Za dane Q i k

- Pročitaj svaku tablicu (u paraleli)
    - Dohvati fingerprintove u Ti čiji se značajni Pi bitovi podudaraju sa značajnim Pi bitovima od Πi(Q)
        - Ti'
        - O(Pi) koraka (binarna pretraga)
    - Za svaki fingerprint u Ti', provjeri je li njegova hamingova udaljenost najviše k bitova od Πi(Q)
- Primjer sa t=20, f=64, k=3, |F| = 8B
    - Podjeli f u 6 blokova (4x11 + 2x10 bitova)
    - Odaberi 3 od 6 blokova (6povrh 3) = 20 putanja
    - Posloži te blokove kao značajne bitove
    - P=zbroj tih bitova
        - 31,32 ili 33
    - U prosjeku upit vraća 2^{34-31}=8 fingerprintova
    - t i p parametri
        - t ~ p
        - vrijeme upita ~ 1/p
        - zahtjevi za pohranu ~ p
        - prostor/vrijeme
            - analitička solucija za t

Skupni upiti koristeći MapReduce i GFS

- F i Q su datoteke u GFS(sa replikacijom)
- F ~ 64GB, Q ~ 8MB
- F je spremljen u GFS komadima
- Broj mapera = broj F komada
- Map:
    - Rješava hamingovu udaljenost za komad (64Mb) i emitira listu sličnih duplikata
- Reduce:
    - Uklanja duplikate