Grafovi za Prepoznavanje i Pretragu Sličnih Entiteta

finbet

1. Uvod

- Potreba za pronalaženjem sličnih entiteta u velikim skupo -vima podataka je prisutna u mnogim oblastima:

1. Pretraživanje teksta
2. Preporučivanje proizvoda
3. Pretraživanje slika
4. Prepoznavanje

- Kako bi se efikasno izvršila pretraga nad velikom količin-om podataka, tokom godina su razvijane različite metode i korišćene različite strukture podataka. Ovde će biti predst avljeni HNSW grafovi i KNN pretraga nad njima.

- Kao primer, biće uzeta pretraga sličnih slika lica.

finbet

Primeri

Samsung Gallery

Na Samsung telefonima, u galeriji je moguće pretražiti slična lica na slikama.

Ostali proizvođači telefona nude slične mogućnosti.

Google Vertex

Google-ov Vertex Matching engine je tehnologija koja se koristi za pretragu vektoriskih prostora i koristi se u servisima kao što su Google pretraga, youtube preporuke, playstore preporuke.

finbet

2. Predstavljanje podataka

Slike lica je prvo potrebno predstaviti u prostoru koji je niže dimenzije, a u kom je moguće zadržati sve informacije koje su neophodne za prepoznavanje.

Moguće ih je predstaviti kao vektore realnih brojeva. Za te potrebe se koriste neuronske mreže koje su obučene da izdvajaju bitne karakteristike slike (eng. feature embeddings).

Za potrebe ovog primera je uzeta varijanta FaceNet-a koja konvertuje sliku u 512-dimenzioni vektor.

$$x_i \in \mathbb{R}^{512}$$

finbet

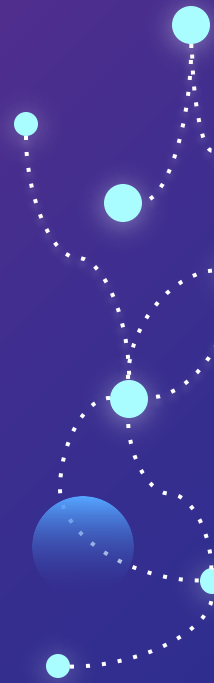
3. KNN pretraga

Algoritmi za pretragu koji se najčešće koriste se oslanjaju na algoritam pretrage K najbližih suseda (KNNS - *K-Nearest Neighbor Search*). KNN pretraga podrazumeva da je definisana funkcija na osnovu koje može da se meri rastojanje između elemenata koji se pretražuju.

Za određivanje distance između vektorskih reprezentacija dve slike lica, najčešće se koristi kosinusna distanca. Kosinusna distanca između dva vektora se računa kao kosinus ugla između njih.

Kosinusna distanca je uvek u intervalu [0, 2], a što je manja, to su slike sličnije. Ukoliko su slike identične, kosinusna distanca je 0, a ukoliko su slike potpuno različite, kosinusna distanca je 2.

Alternativno, može se koristiti i L_2 norma.



finbet

Kosinusna distanca

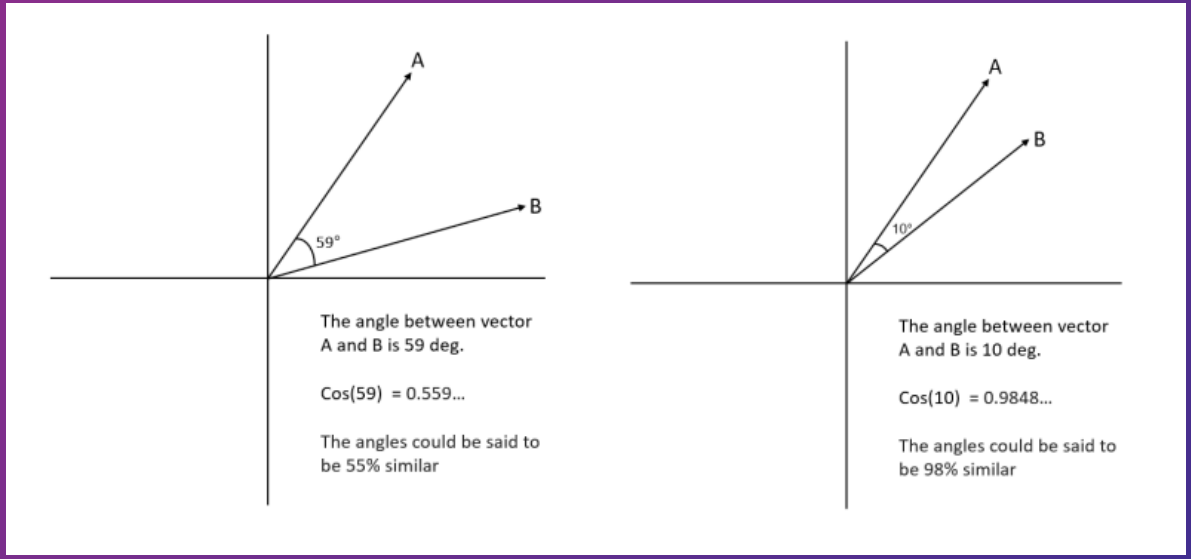
Definisana je kao:

$$
\begin{aligned}
cos\_distance &= 1 - cos\_similarity \\
&= 1 - cos(\theta) \\
&= 1 - \frac{A \cdot B}{||A||||B||} \\
&= 1 - \frac{\sum_{i=1}^{n} A_i B_i}{\sqrt{\sum_{i=1}^{n} A_i^n} \sqrt{\sum_{i=1}^{n} B_i^n}}
\end{aligned}
.
$$

gde su A i B vektori koji odgovaraju slikama koje se porede, a θ ugao između njih.

finbet

Kosinusna distanca



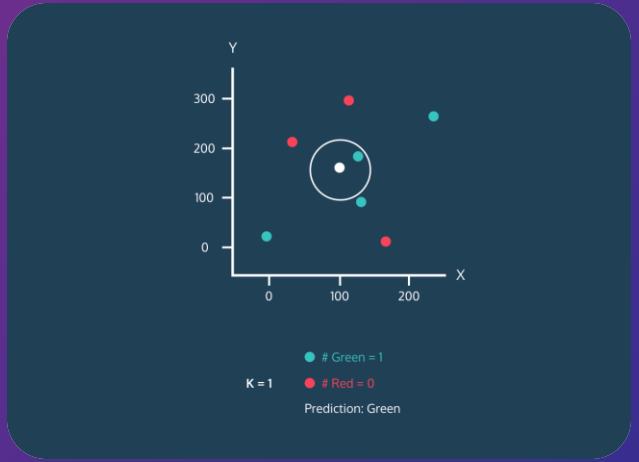
Levo: ugao između vektora je veći, što znači da su oni međusobno dalji i da su slike kojima odgovaraju manje slične.

Desno: ugao između vektora je manji, što znači da su oni međusobno bliži i da su slike kojima odgovaraju sličnije.

finbet

KNN pretraga

- Naivni pristup KNN pretrage se zasniva na tome da se za svaki element iz skupa podataka izračuna rastojanje od svih ostalih elemenata, da se zatim izabere K elemenata koji su najbliži da tom elementu.
- Nažalost, složenost ovog pristupa raste linearno sa porastom b roja elemenata u skupu podataka, čineći ga neupotrebljivim za realne primene.
- Zbog toga se koriste različite strukture podataka koje omogu ćavaju bržu pretragu, kao i aproksimacije pretrage.



Gif from ***eunsukim.me***

finbet

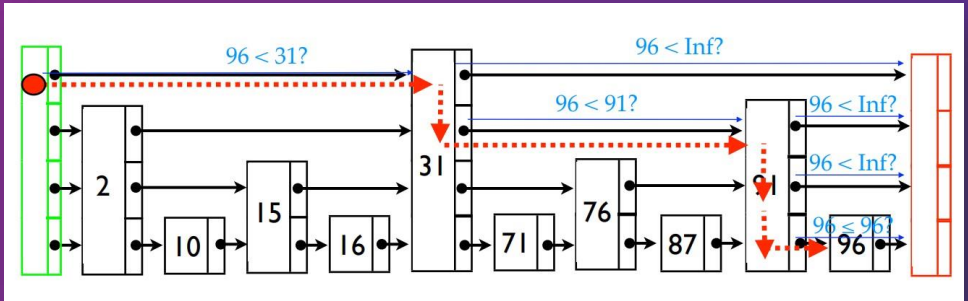
4. Aproksimativna KNN pretraga pomoću HNSW grafova

- HNSW (Hierarchical Navigable Small World) graf predstavlja state-of-the-art strukturu podataka za aproksimativnu KNN pretragu.
- HNSW graf je struktura podataka koja se sastoji iz više nivoa.
- HNSW uzima koncept pretrage od skip listi.

finbet

4.1. *Skip* lista

- *Skip* lista je struktura podataka koja omogućava brzu pretragu, a sastoji se od više nivoa.
- Viši nivoi sadrže manje elemenata, između kojih su uspostavljene duže konekcije. Kako se spuštamo na niže nivoe, broj elemenata raste, a konekcije postaju kraće. Najniži nivo sadrži sve elemente originalne liste.
- Pretraga za nekim elementom *k* počinje od najvišeg nivoa. Kada nađemo element koji je veći od *k*, vraćamo se na prethodni manji element, spuštamo se na niži nivo, i nastavljamo pretragu od tog elementa.



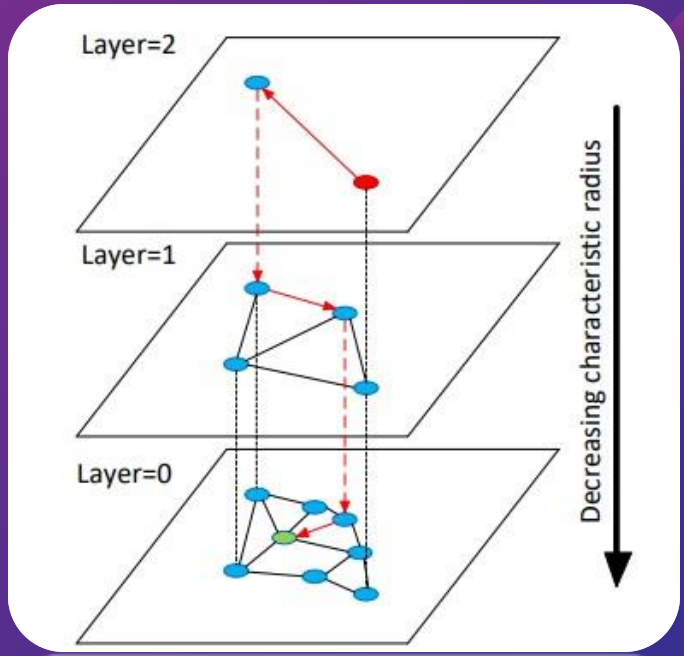

```
1: If k = key: done;
2: k < next key: go down a level;
3: k ≥ next key: go right;
```

Pseudokod za pretragu

finbet

4.2. HNSW graf

- Sličan koncept se primenjuje i pri kreiranju i pretrazi HNSW grafa. On se takođe sastoji iz više nivoa, s tim što se na svakom nivou umesto liste nalazi graf.
- Čvorovi grafa predstavljaju *feature* vektore koji odgovaraju slikama lica. Grane kojima su čvorovi povezani odgovaraju kosinusnim distancama između tih vektora.
- Najviši nivo ima najmanje čvorova i najduže veze između njih, dok svaki naredni sloj ima sve više čvorova i sve kraće veze.



4.2.1. Pretraga HNSW grafa

KORAK 1

Pretraga počinje od slučajno odabranog čvora na najvišem sloju grafa.

KORAK 2

Trenutni sloj se pretražuje sve dok se ne pronađe lokalni minimum – čvor koji ima najmanju distancu od traženog čvora. Odnosno, dok se ne pronađe traženi broj najbližih čvorova. Pretraga sloja se vrši heuristički. Prilikom pretrage se čuva dinamička lista najbližih suseda. Lista se u svakom koraku ažurira na osnovu evaluacije suseda čvorova koji su prethodno dodati u nju. Kada lista dostigne maksimalnu veličinu, ako se naiđe na element koji je bliži traženom čvoru od najdaljeg elementa liste, taj najdalji element će biti zamenjen njime. Kada se evaluiraju svi susedi svakog elementa liste, pretraga sloja se završava.

KORAK 3

Nakon što se pronađe lokalni minimum, pretraga se nastavlja na narednom (nižem) sloju, od čvora koji je predstavljao lokalni minimum na prethodnom sloju.

KORAK 4

Postupak se ponavlja sve dok se ne dođe do najnižeg sloja, kada se vraća *K* suseda najbližih traženom čvoru.

finbet

4.2.2. Kreiranje HNSW grafa

KORAK 1

Za svaki element koji se dodaje se prvo odredi maksimalni sloj *l* na kom će se nalaziti. To se radi probabilistički – *l* se bira iz eksponencijalno opadajuće raspodele (normalizovane parametrom koji određuje maksimalan broj konekcija po čvoru).

KORAK 2

Zatim sledi pretraga za najbližim susedima elementa *q* koji se dodaje.

KORAK 3

Prva faza pretrage ide od najvišeg sloja ka odabranom sloju *l*. Ovime se dobija inicijalni element koji je najbliži čvoru *q*.

KORAK 4

U sledećoj fazi pretrage se polazi od sloja odabranog sloja *l* (ako je on ispod najvišeg sloja) i ide ka nižim slojevima, pri čemu pretraga počinje od čvora pronađenog u prethodnom koraku. (Ako se desi da je *l* iznad najvišeg sloja, čvor *q* se dodaje kao polazni čvor hnsw-a).

KORAK 5

Ovime se u svakom sloju pronalazi definisani broj najbližih suseda čvoru *q*.

KORAK 6

Zatim se kreiraju grane od *q* ka pronađenim susedima u trenutnom sloju (pri čemu se vodi računa o maksimalnom dozvoljenom broju konekcija koje *q* može da ima).

KORAK 7

Pretraga u narednom sloju se nastavlja počevši od najbližih suseda pronađenih u prethodnom nivou, sve dok se ne stigne do najnižeg sloja.

finbet

5. Alternativne metode

Annoy

Annoy metoda koristi binarna stabla pretraživanja. Mnogo puta deli prostor podataka i gleda samo njegov deo kako bi pronašla bliske susede.

Inverted file index - IVF

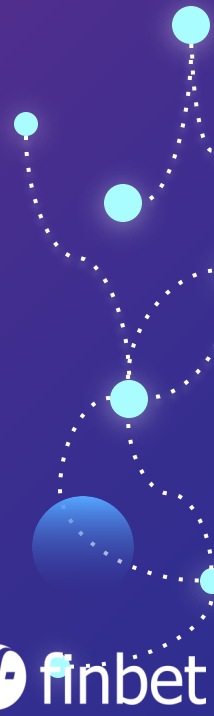
IVF metoda smanjuje vreme pretrage tako što ceo skup podeli u manje particije, gde svakoj particiji asocira centroid.

Locality Sensitive Hashing – LSH

Ideja sa ovom metodom je da sve iste ili slične vektore hešira u istu vrednost.

Product Quantization

Slično kao metode kvantizacije, i ova metoda kompresuje vektore u manji prostor, i na taj način čuva memoriju.



finbet

Hvala na pažnji!

finbet