



Tema:

# PRETRAGA MULTIMODALNIH VEKTORSKIH PODATAKA

Master rad

Studijski program: Veštačka inteligencija  
i mašinsko učenje

Mentor:

Aleksandar Stanimirović

Student:

Filip Trajković 1574

# SADRŽAJ

## [1] Pojam vektorskih podataka



[2]

Pretraga  
vektorskih podataka

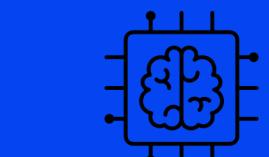
[3]

Vektorske  
baze podataka



[4]

Multimodalni  
modeli



## [5] Aplikacija za vektorskiju pretragu kućnih ljubimaca



## [6] Zaključak



# [1] Pojam vektorskih podataka

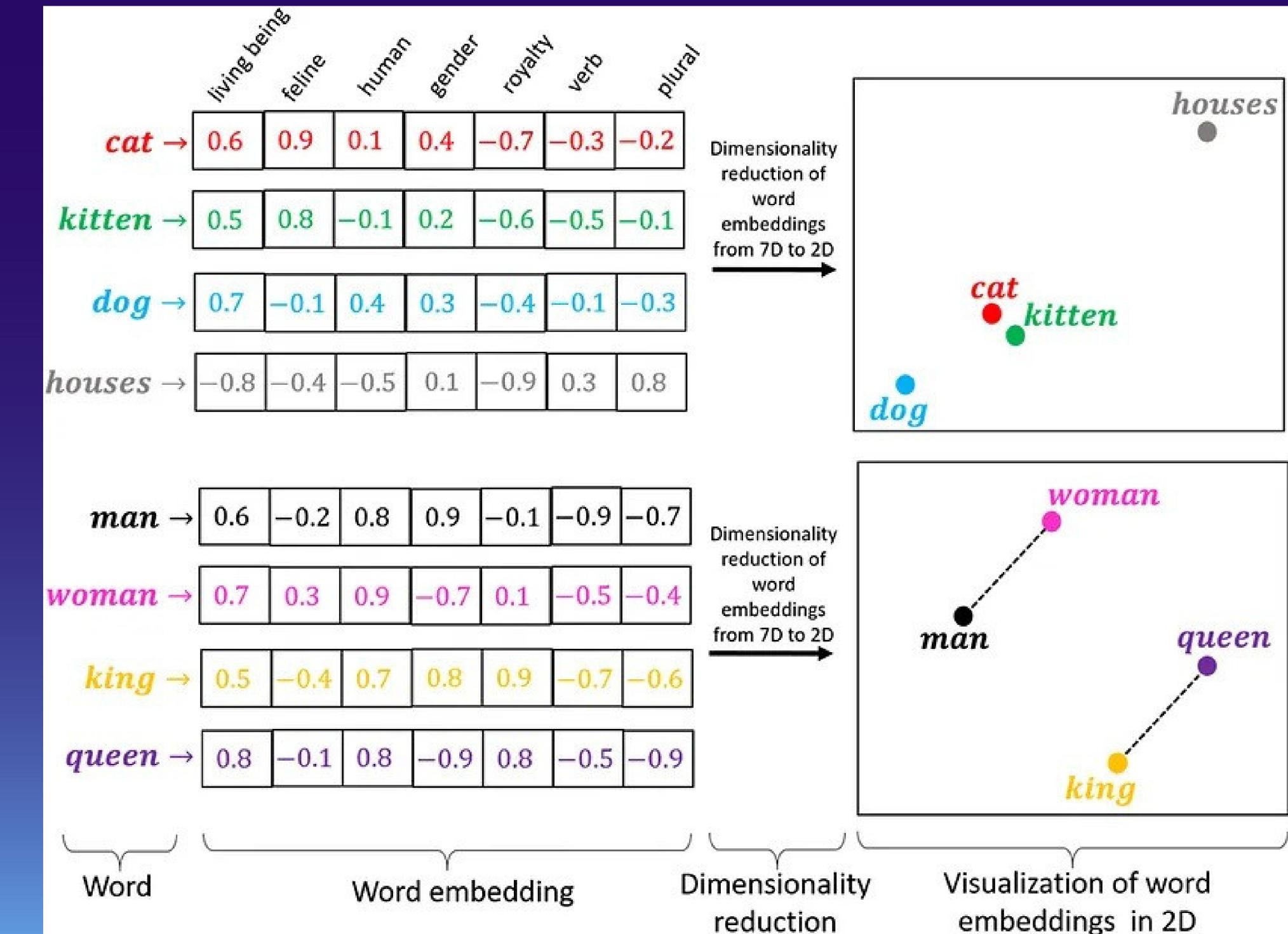
---

**Pojam vektorskih podataka u veštačkoj inteligenciji** odnosi se na podatke predstavljene u višedimenzionalnom prostoru podataka pri čemu se podaci predstavljaju višedimenzionalnim vektorima kod kojih svaki element vektora nosi kvalitativnu kontekstualnu informaciju o datom podatku.

# [1] Pojam vektorskih podataka

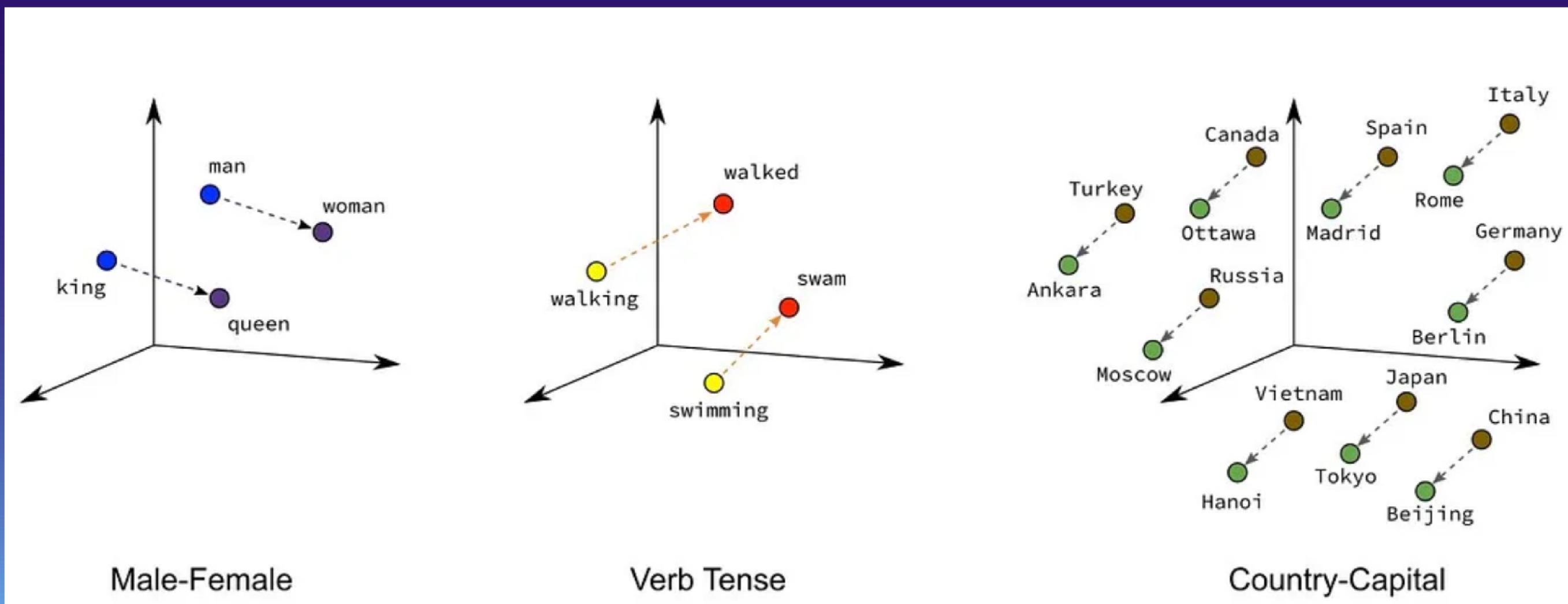
Procesima generisanja vektora na osnovu ulaznih nestruktuiranih podataka vrši se kreiranje struktuiranih podataka u formatu vektora.

Dobijeni vektorski podaci moraju da očuvaju kontekstualne i semantičke zavisnosti koje važe između ulaznih podataka.

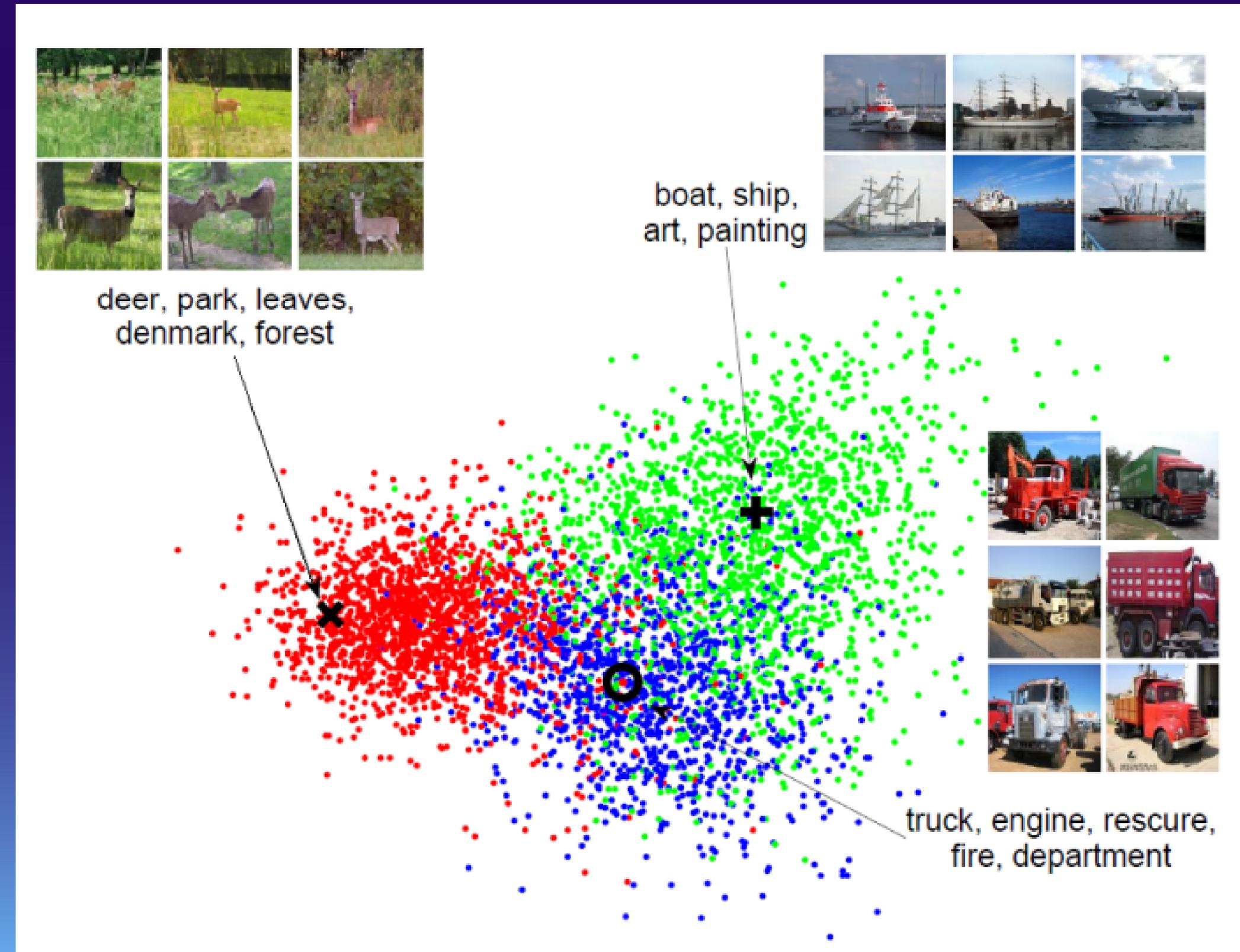


# [1] Pojam vektorskih podataka

Međusobne zavisnosti mogu važiti između različitih vrsta reči poput imenica, glagola i sl. Pozitivne ili negativne zavisnosti mogu postojati između sinonima, homonima ili na osnovu leksičkih pravila jezika.



# [1] Pojam vektorskih podataka



[2]

# Pretraga vektorskih podataka

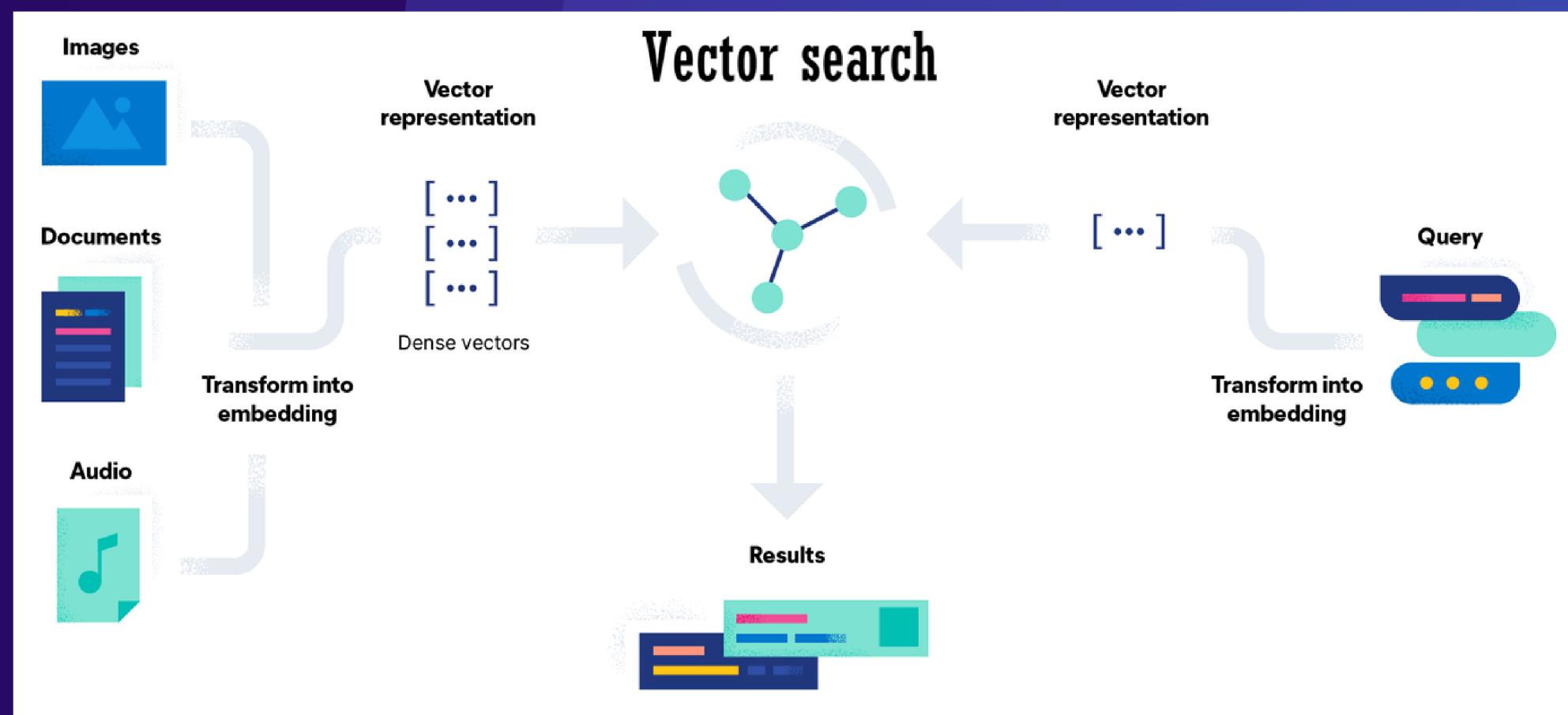
---

Pretraga vektorskih podataka

[2]

# [2] Pretraga vektorskih podataka

Pretraga vektorskih podataka predstavlja složen proces koji obuhvata tehnike usmerene na proces pretrage sličnih vektora u odnosu na prosleđeni vektor (Eng. query vector) pri čemu treba voditi računa o efikasnosti i brzini obrade operacija pretrage.



# [2] Pretraga vektorskih podataka

---

## Metrike sličnosti vektorske pretrage

Metrike sličnosti se primenjuju nad vektorima u konačnom skupu baze vektora i predstavljaju uporednu kvalitativnu meru kojom se vektori iz baze upoređuju sa unetim vektorom sa kojim se vrši upoređivanje.

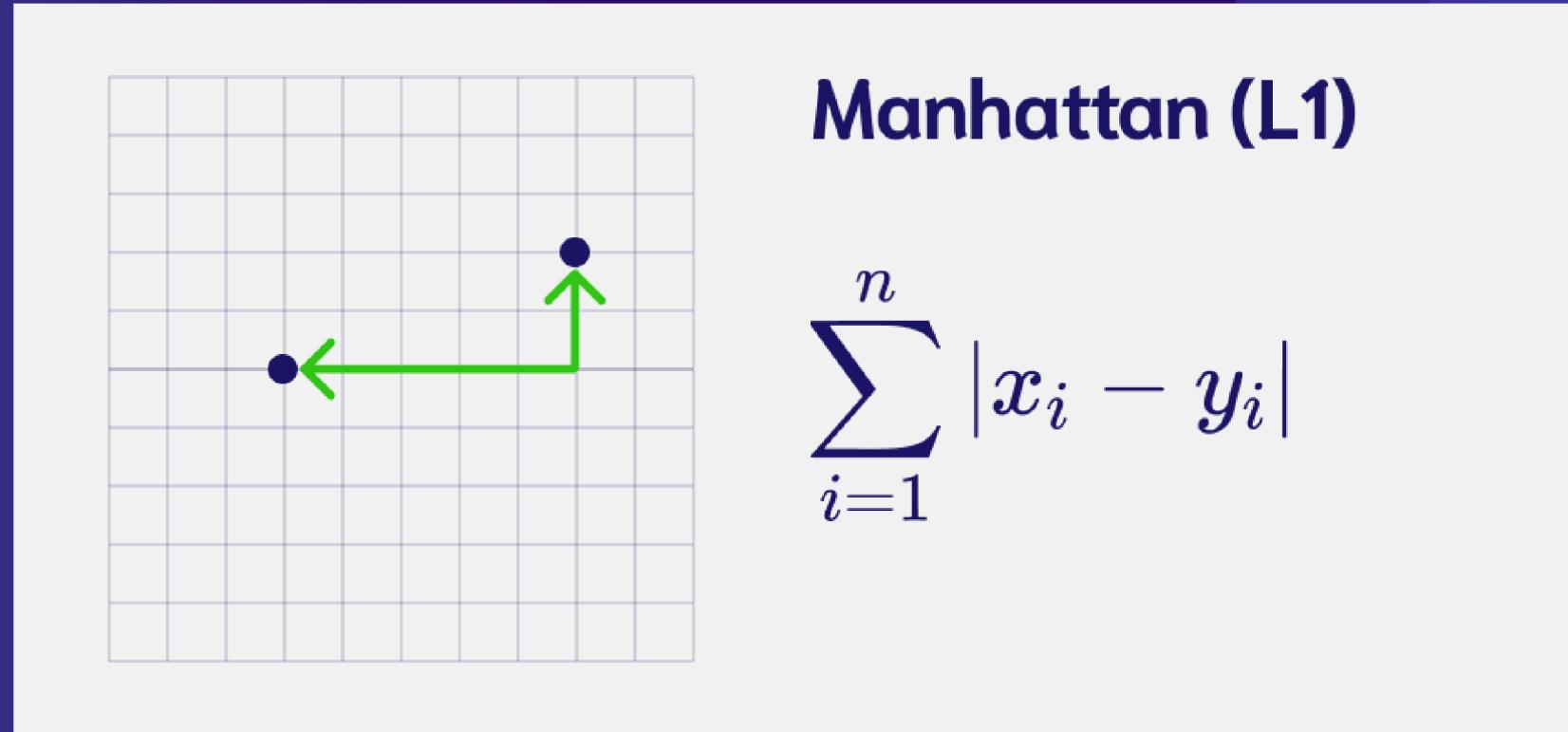
Najučestalije metrike koje se implementiraju u rešenjima pretrage su:

- *Manhattan distanca (L1)*
- Euklidska distanca (L2)
- Skalarni proizvod
- Kosinusna sličnost

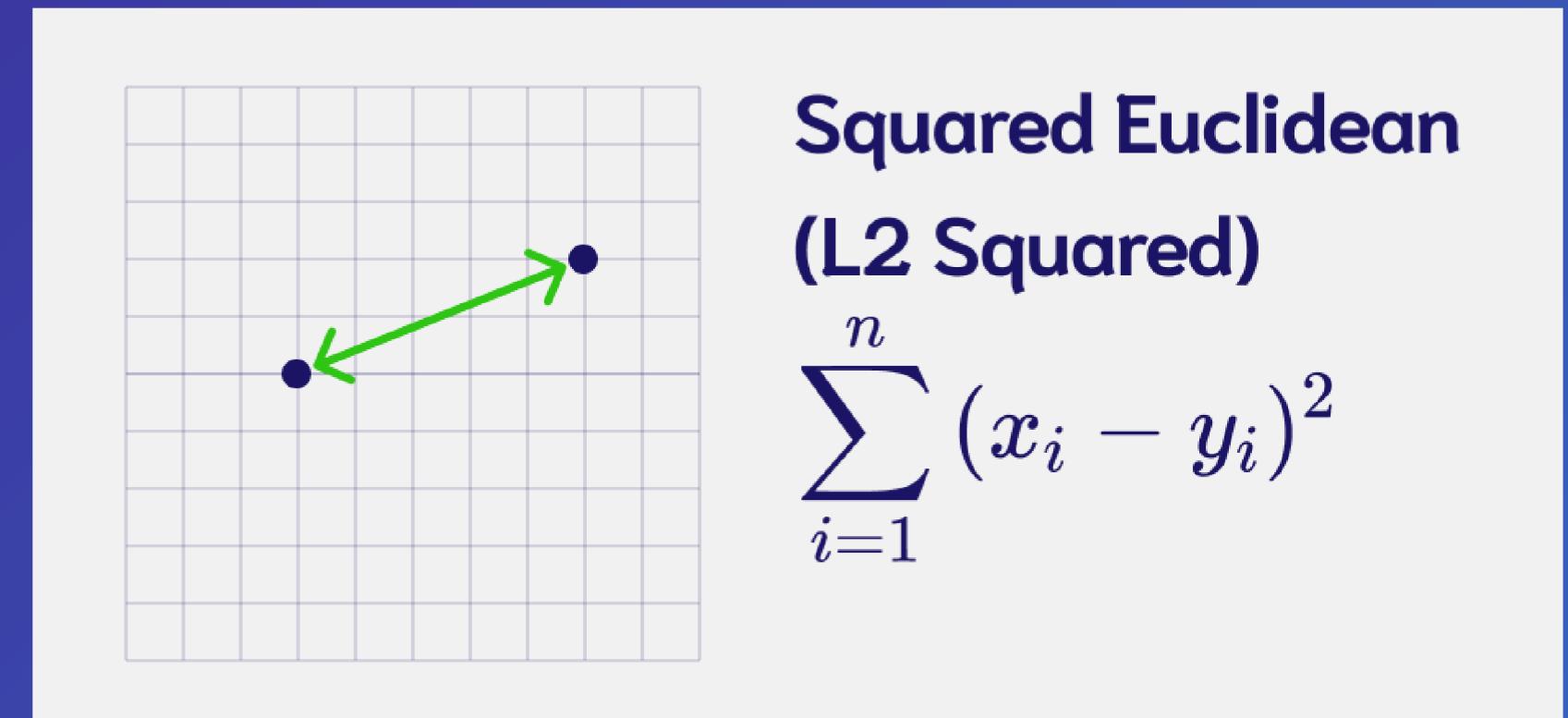
# [2] Pretraga vektorskih podataka

## Metrike sličnosti vektorske pretrage

### Manhattan distanca



### Euklidska distanca

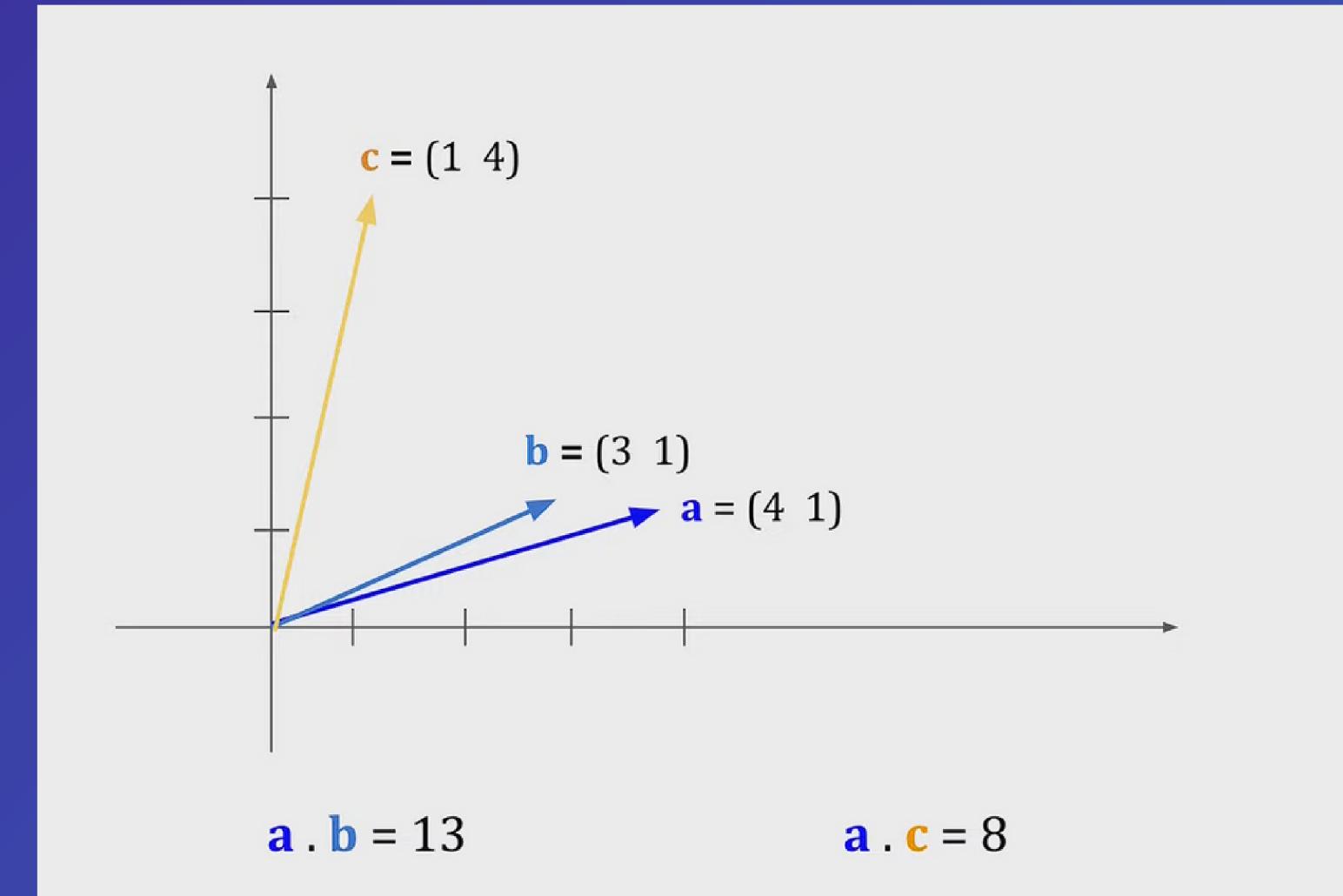


# [2] Pretraga vektorskih podataka

## Metrike sličnosti vektorske pretrage

### Skalarni proizvod

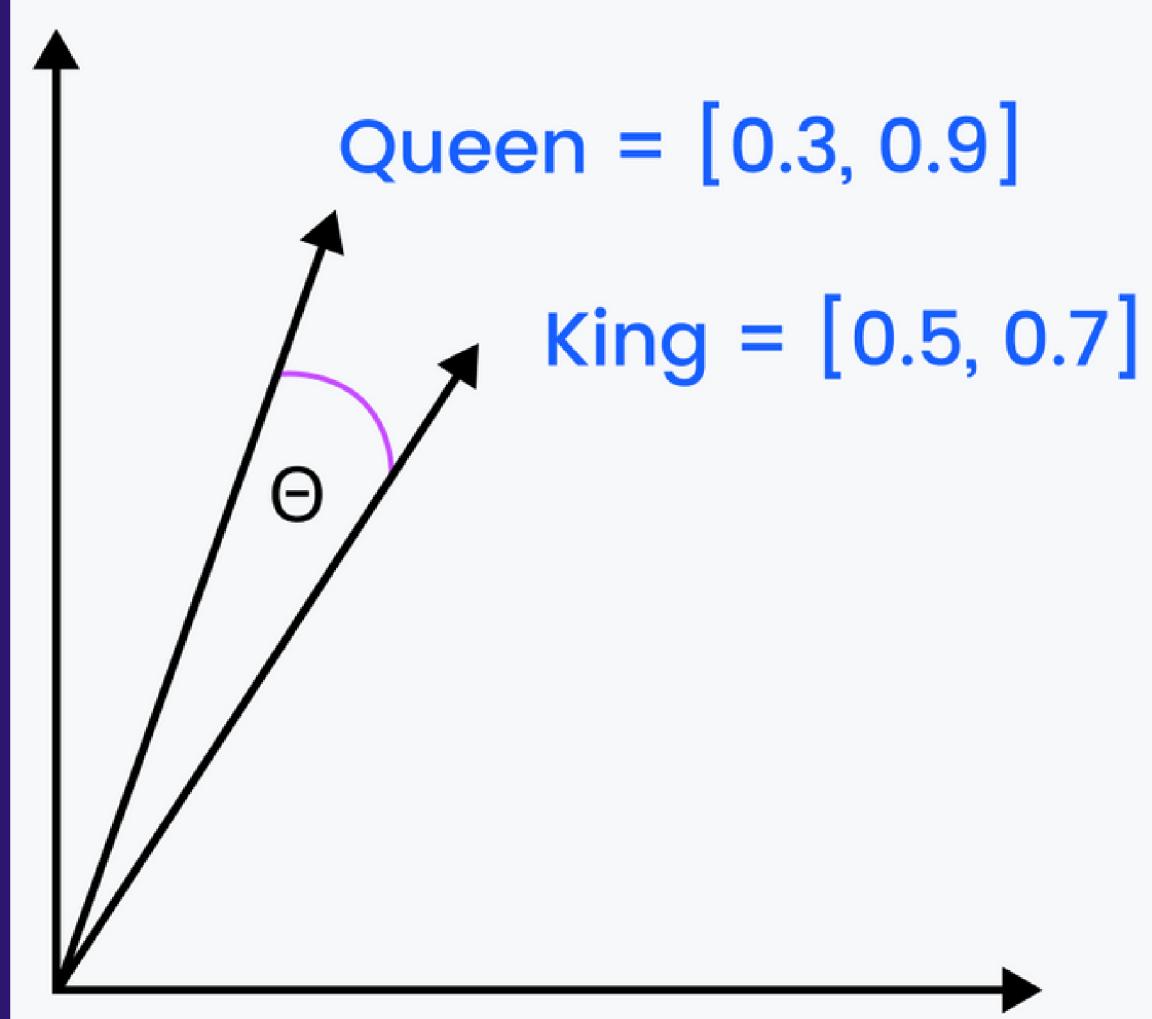
$$\mathbf{a} \cdot \mathbf{b} = \sum_{i=1}^n a_i b_i = a_1 b_1 + a_2 b_2 + a_3 + b_3 + \dots + a_n b_n$$



# [2] Pretraga vektorskih podataka

## Metrike sličnosti vektorske pretrage

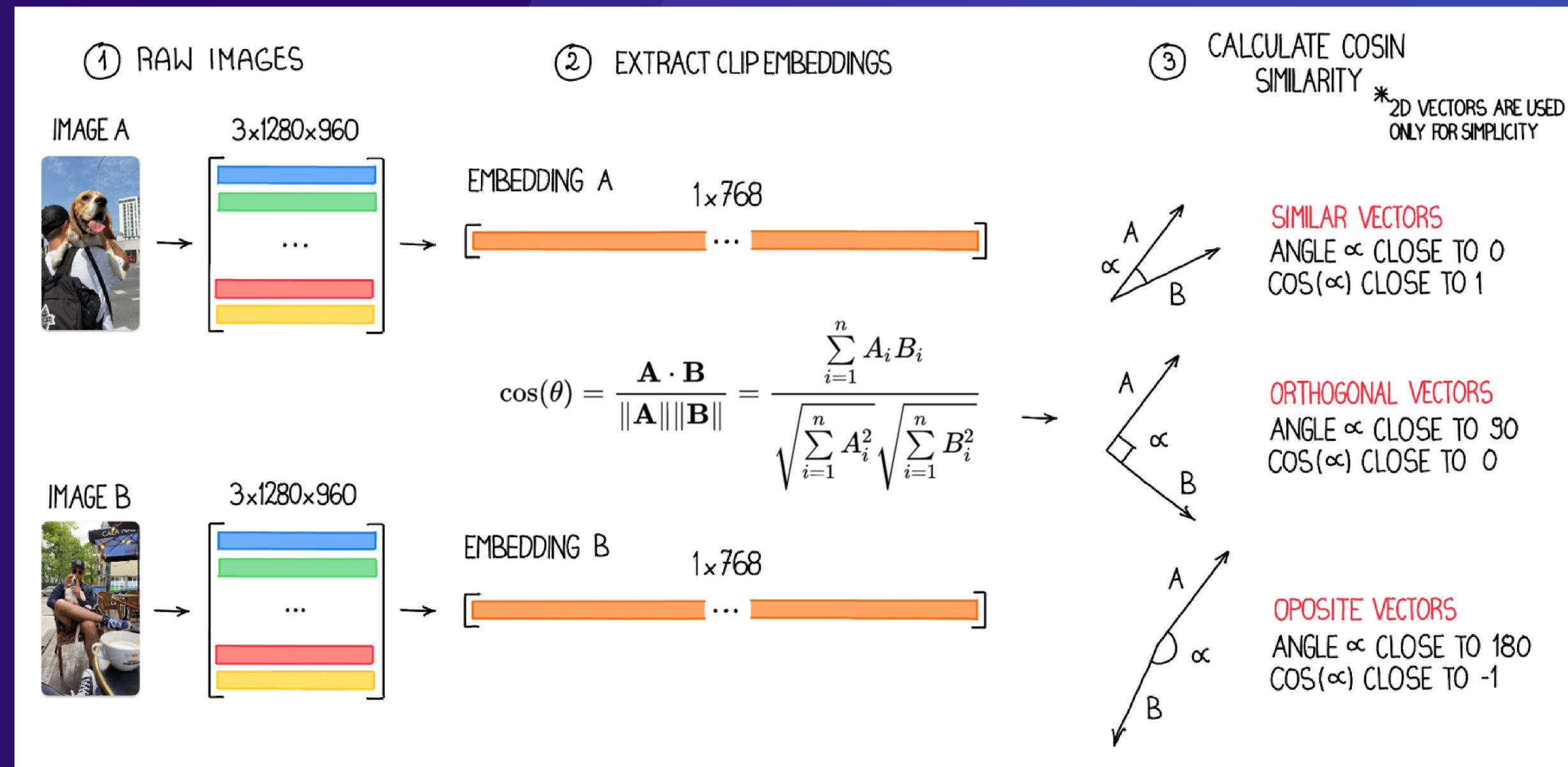
### Kosinusna sličnost



$$\begin{aligned}\text{Cos(Queen, King)} &= \frac{(0.3*0.5)+(0.9*0.7)}{\sqrt{0.3^2+0.9^2} * \sqrt{0.5^2+0.7^2}} \\ &= \frac{0.15+0.63}{\sqrt{0.9^2} * \sqrt{0.74}} \\ &= \frac{0.78}{\sqrt{0.666}} \\ &= 0.97\end{aligned}$$

# [2] Pretraga vektorskih podataka

## Metrike sličnosti vektorske pretrage Kosinusna sličnost - primer korišćenja



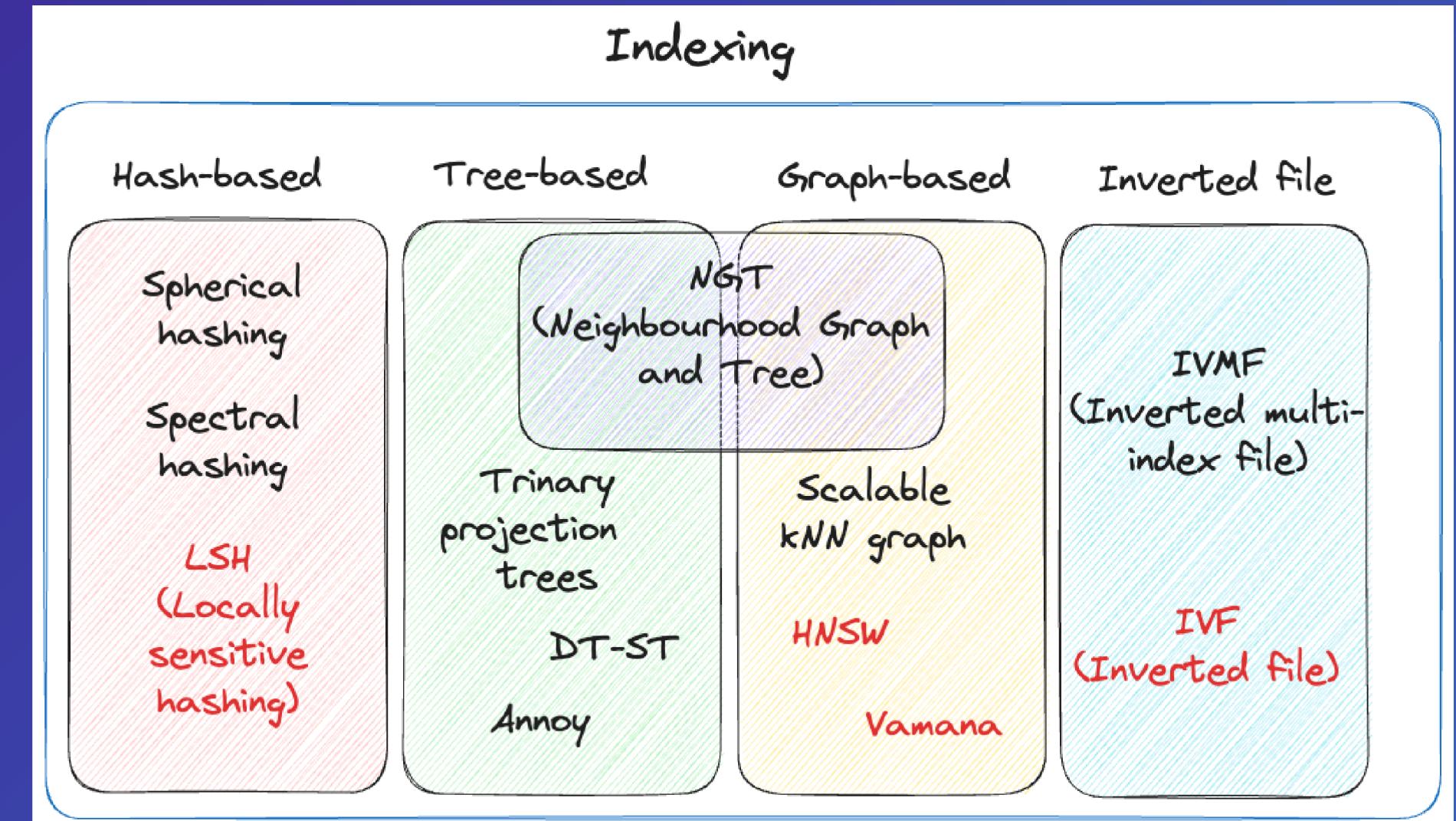
# [2] Pretraga vektorskih podataka

## Indeksiranje vektora

Indeksiranje vektora - proces kreiranja sistema za brzo i efikasno manipulisanje vektorima

Kreira se jedinstvena struktura koja se naziva indeks vektora.

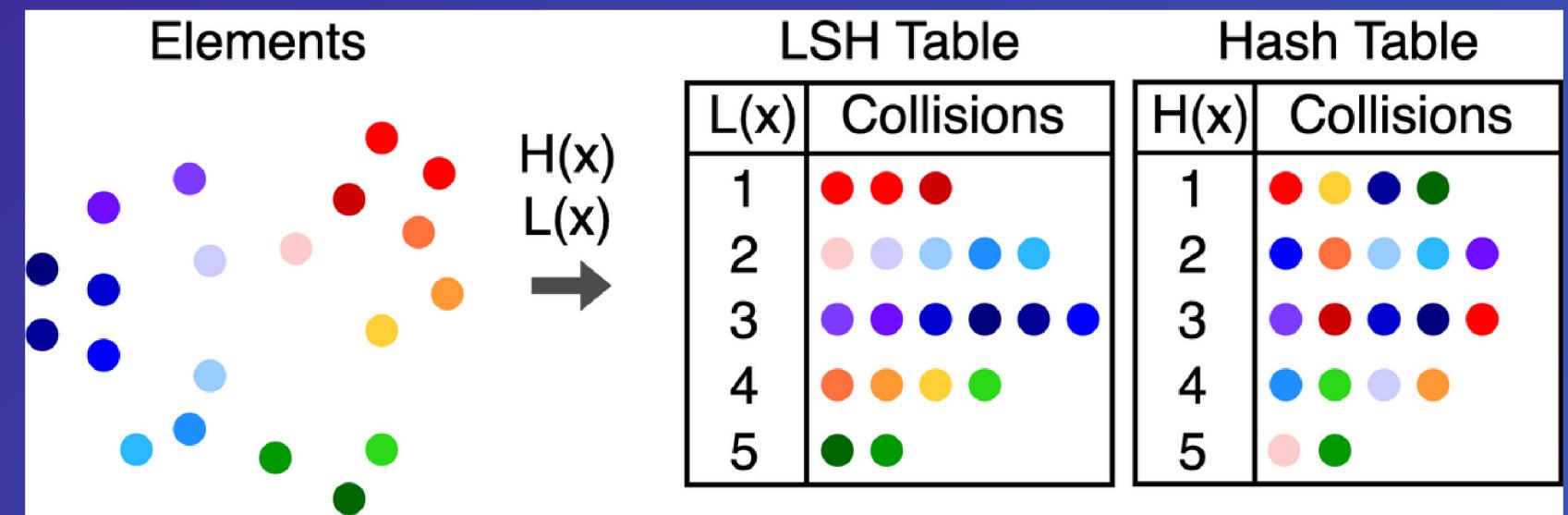
Indeks vektora predstavlja organizovanu strukturu za pretragu sličnih vektori i može biti baziran na stablima, heš-tabelama, grafovima ili invertovanim fajl indeksima.



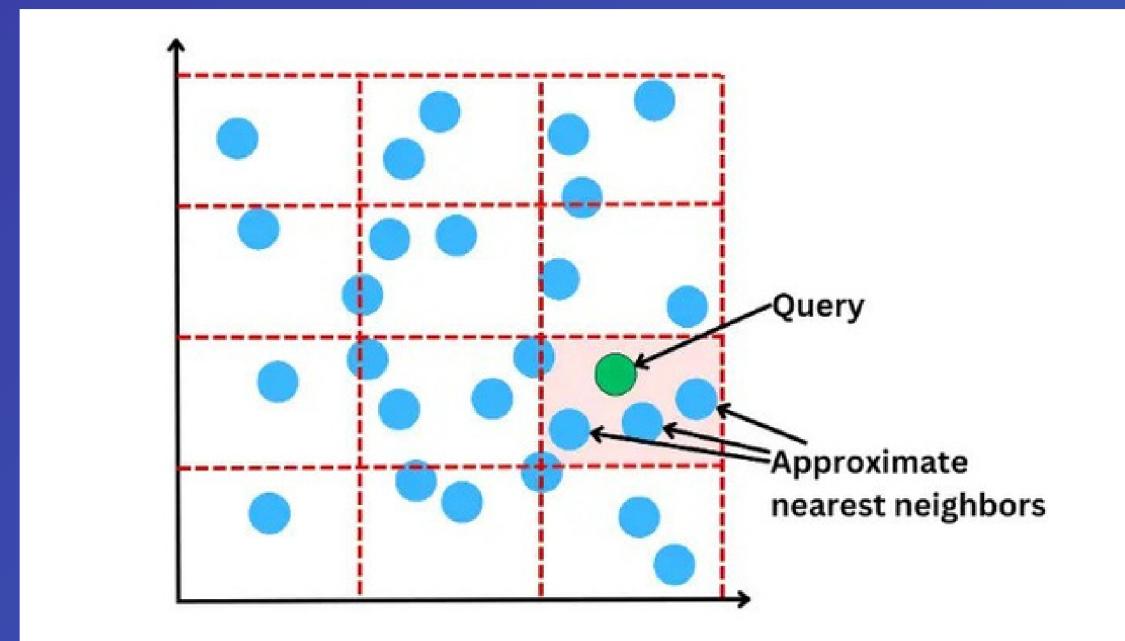
# [2] Pretraga vektorskih podataka

## Indeksiranje vektora - Hash-based

**LSH (Eng. Locally sensitive hashing)**  
- proces kod kojeg se vektori procesiraju kroz heš-funkciju i odvajaju u nezavisne grupe.



Bliski vektori u vektorskem prostoru se odvajaju u odeljke u kojima se potom vrši pretraga sličnih vektora sa ulaznim koji je takođe heširan i prosleđen samo jednoj grupi.

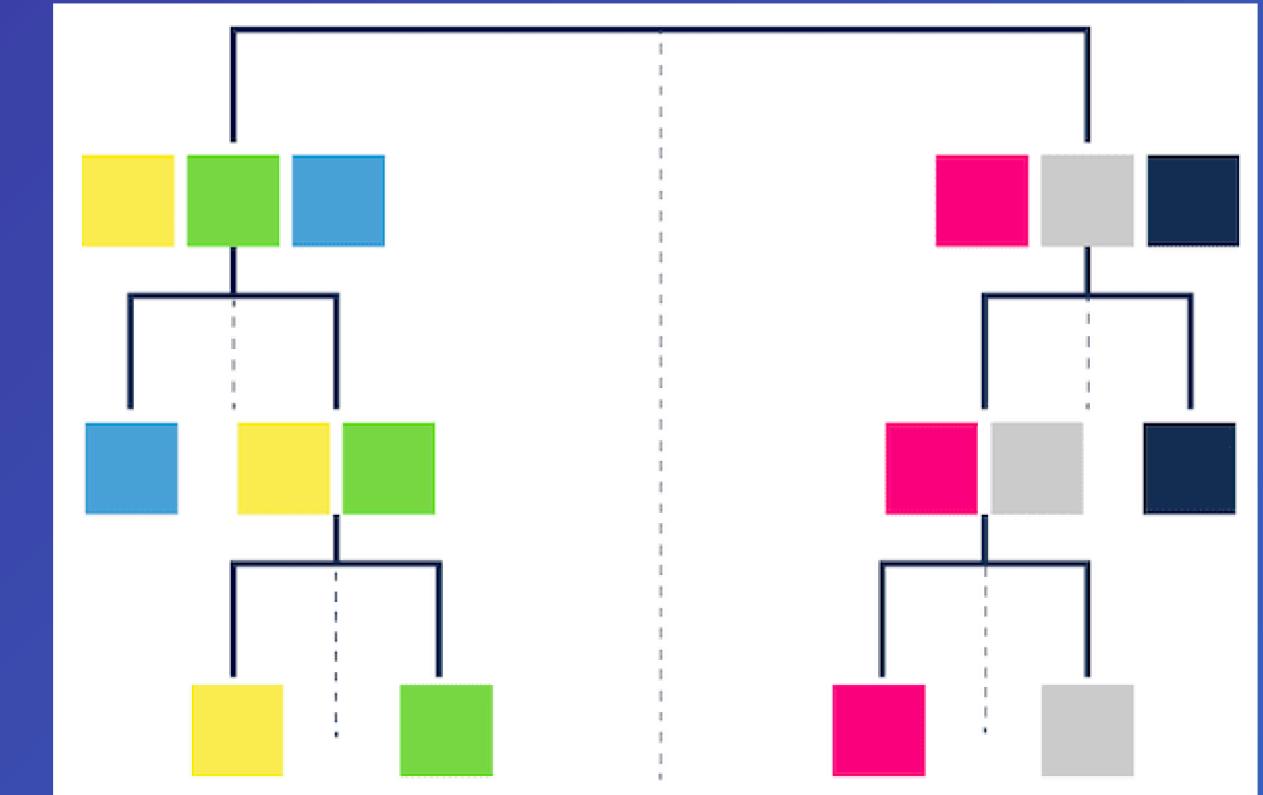
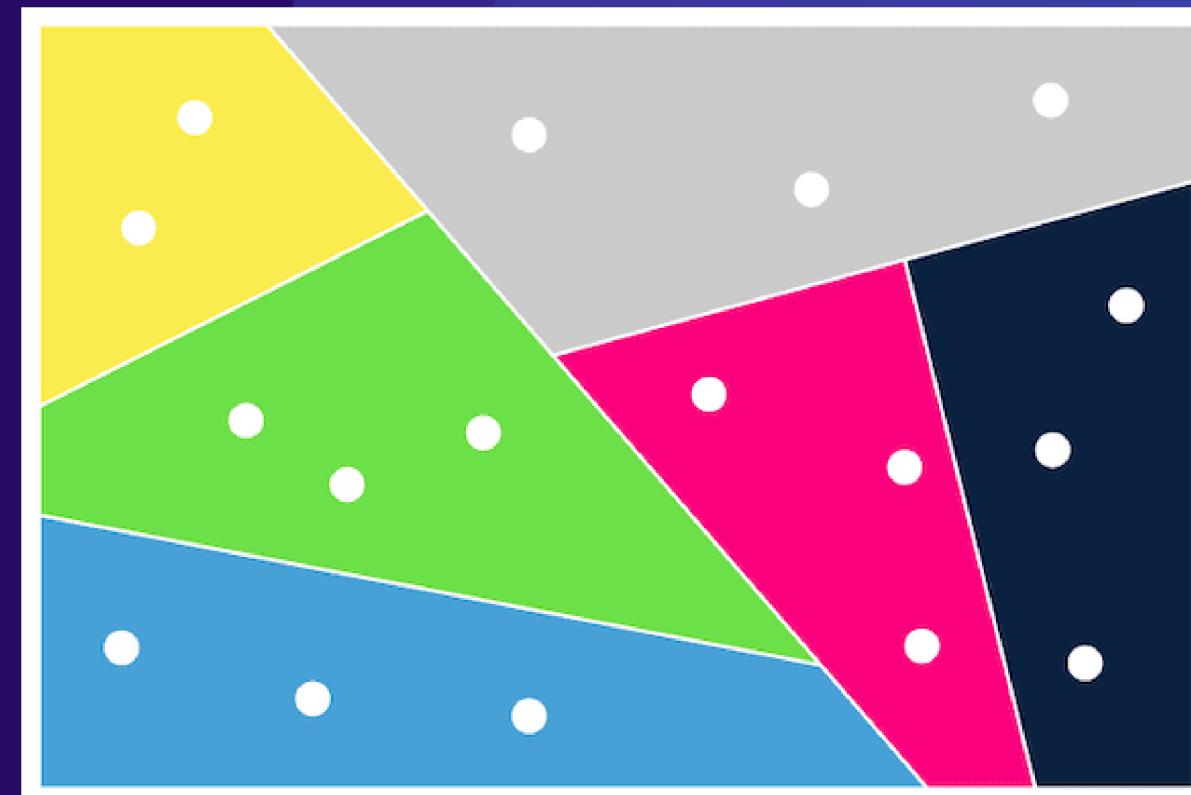


# [2] Pretraga vektorskih podataka

## Indeksiranje vektora - Tree-based

Kod Tree-based algoritama vektori se dele po granama stabla prema sličnosti, dok se prilikom kreiranja indeksa definiše metrika sličnosti i broj maksimalnih stabala koje je moguće kreirati.

Stabla grananja se formiraju na osnovu linearnih funkcija preseka vektorskog prostora.

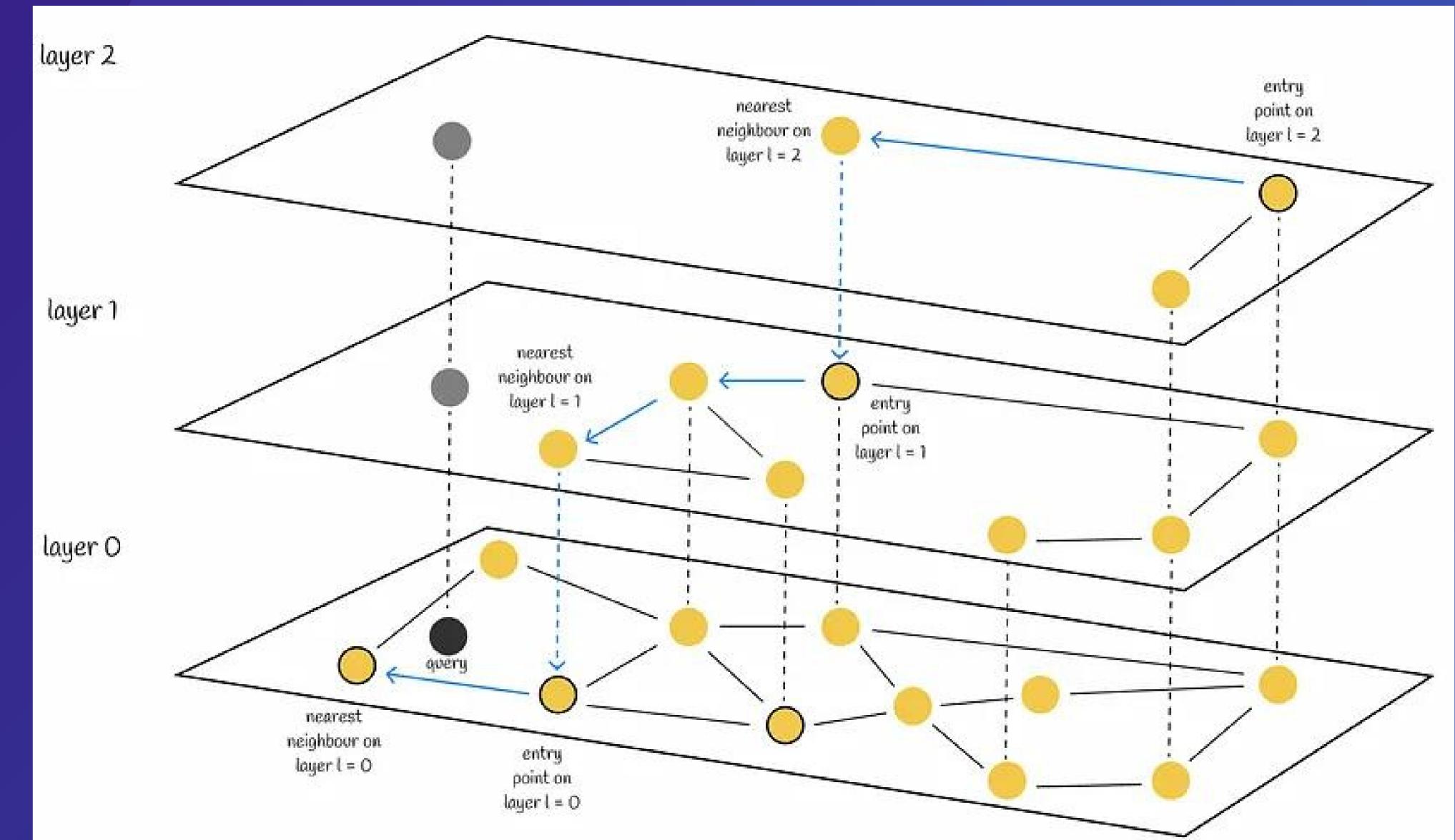


# [2] Pretraga vektorskih podataka

## Indeksiranje vektora - Graph-based

Indeksiranje bazirano na grafovima predstavlja pristup kod kojih se vektori predstavljaju čvorovima grafova, pri čemu veze između čvorova predstavljaju veze između sličnih vektora.

Obilazak sličnih vektora se vrši putem njihovih suseda koristeći višeslojnu arhitekturu kod koje su vektori nasumično razbacani po slojevima.

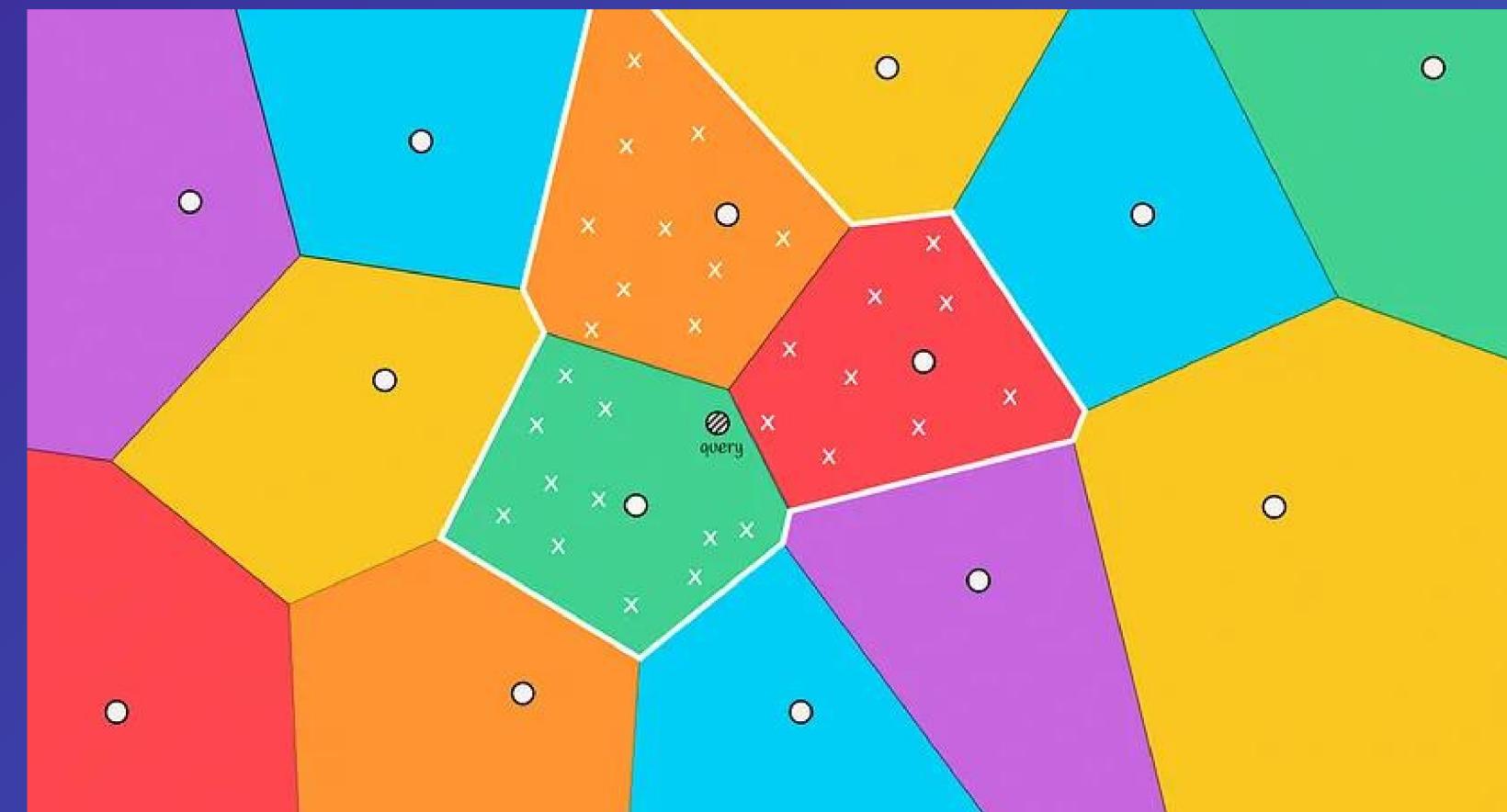


# [2] Pretraga vektorskih podataka

## Indeksiranje vektora - IVF + PQ

Indeksiranje bazirano na invertovanim fajlovima predstavlja metodu kod koje se vektorski prostor deli na *Voronoi* ćelije sličnih vektora pri čemu se tokom pretrage izdvajaju ćelije kandidati i njihovi elementi.

Nad izdvojenim kandidatima se dalje vrši proces *Product Quantization*.

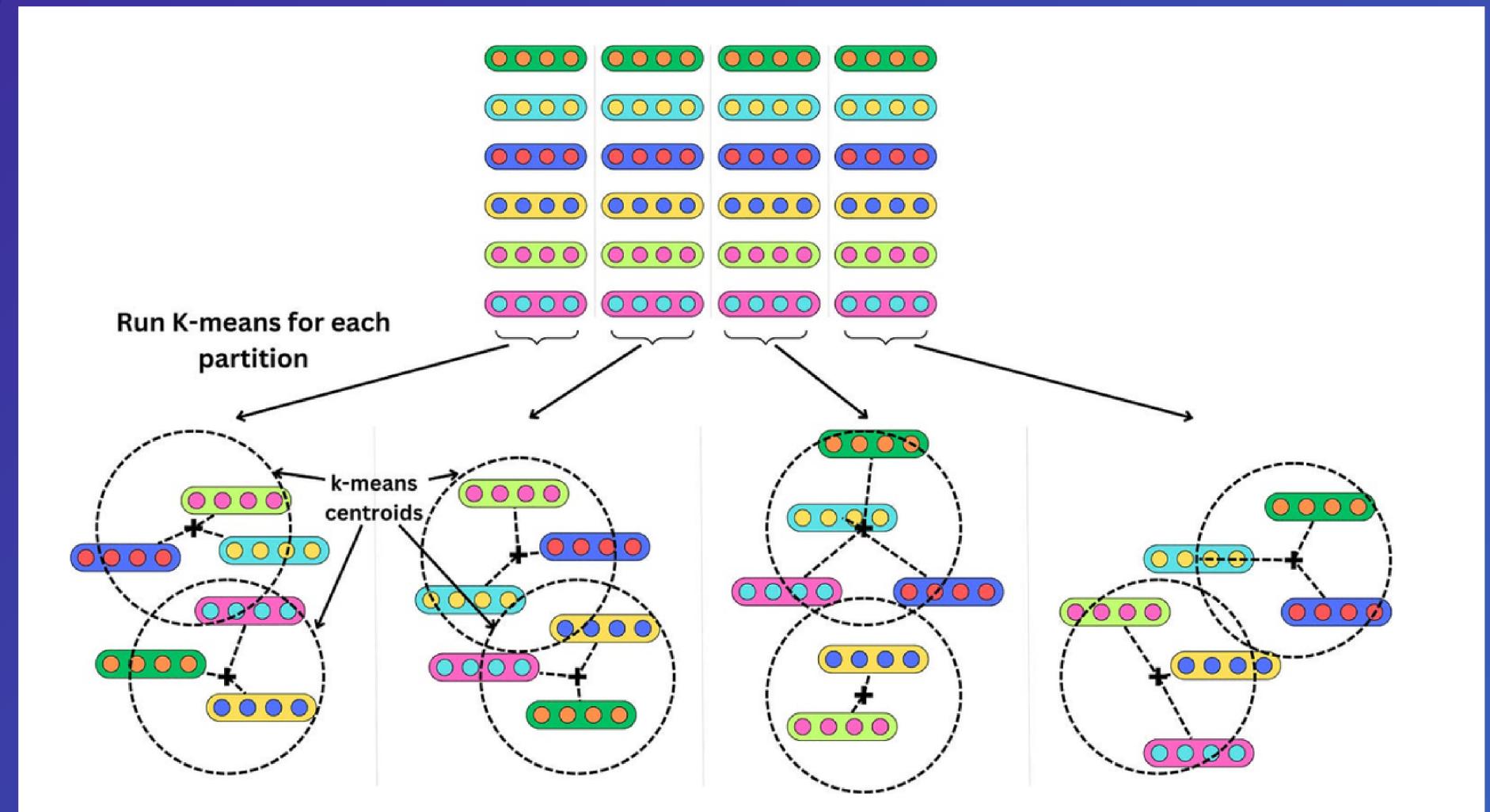


# [2] Pretraga vektorskih podataka

## Indeksiranje vektora - IVF + PQ

Metodom kvantizacije vektora vrši se kompresija veličine i podela na manje celine - particije. Na osnovu izdvojenih celina vrši se klasterizacija pri čemu se vrši izdvajanje centroida.

Svaka particija se predstavlja korespondentnim centroidom, dok se celokupan vektor predstavlja listom centroida.



## [2] Pretraga vektorskih podataka

---

Dosadašnji razvijeni pristupi za manipulacije vektorima poput smeštanja i pretrage vektora:

- Biblioteke za pretragu vektora
- Tradicionalne baze podataka sa ugrađenom podrškom za vektore
- Vektorske baze podataka

# [2] Pretraga vektorskih podataka

## Biblioteke za pretragu vektora

Biblioteke za pretragu vektora predstavljaju biblioteke u Python programskom okruženju kreirane za potpunu manipulaciju operacijama za dodavanje i pretragu vektora bez potrebe za korišćenjem nekih udaljenih servisa.

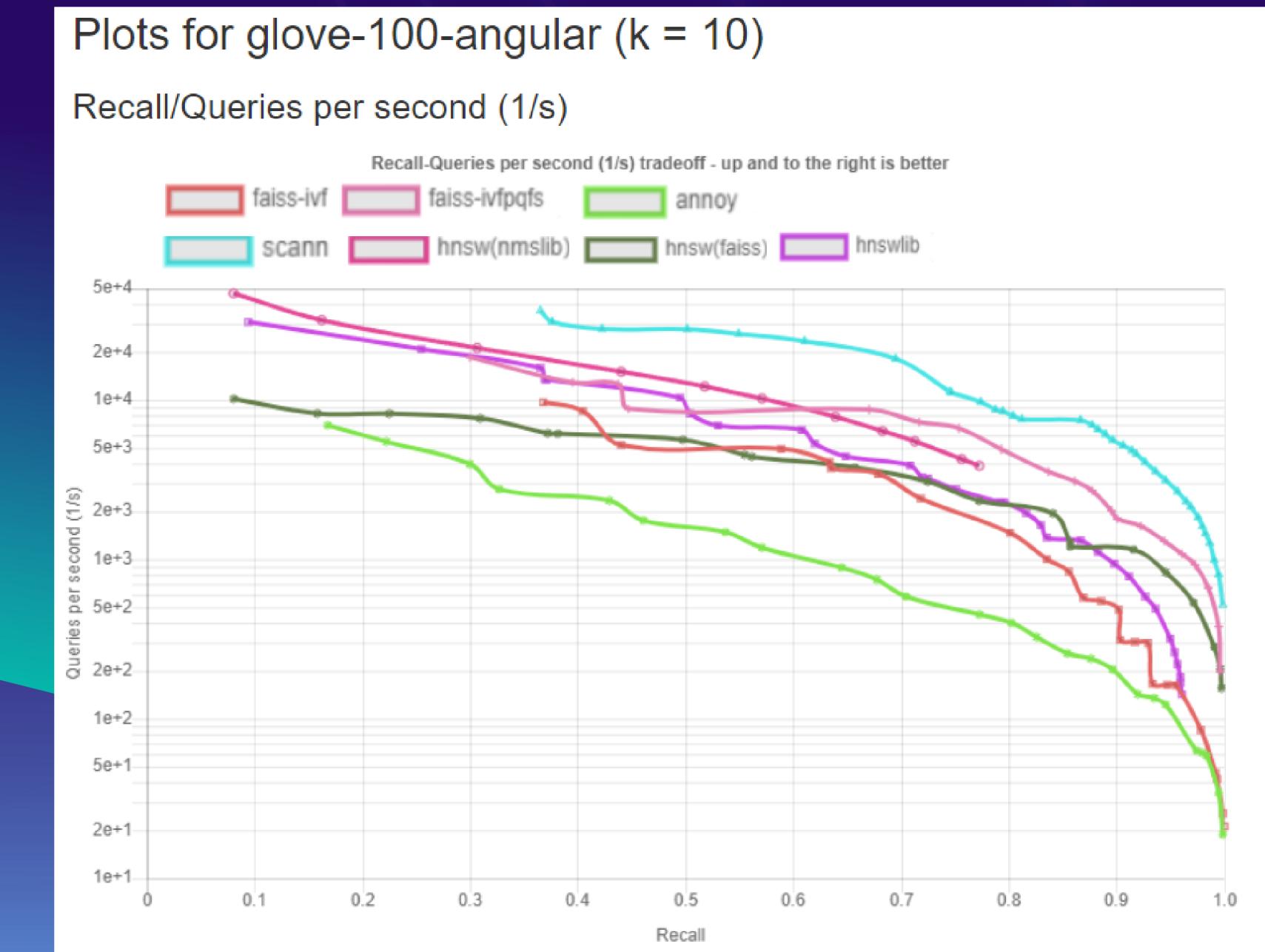
### Predstavnici:

- ANNOY
- FAISS
- ScaNN

# [2] Pretraga vektorskih podataka

## Biblioteke za pretragu vektora

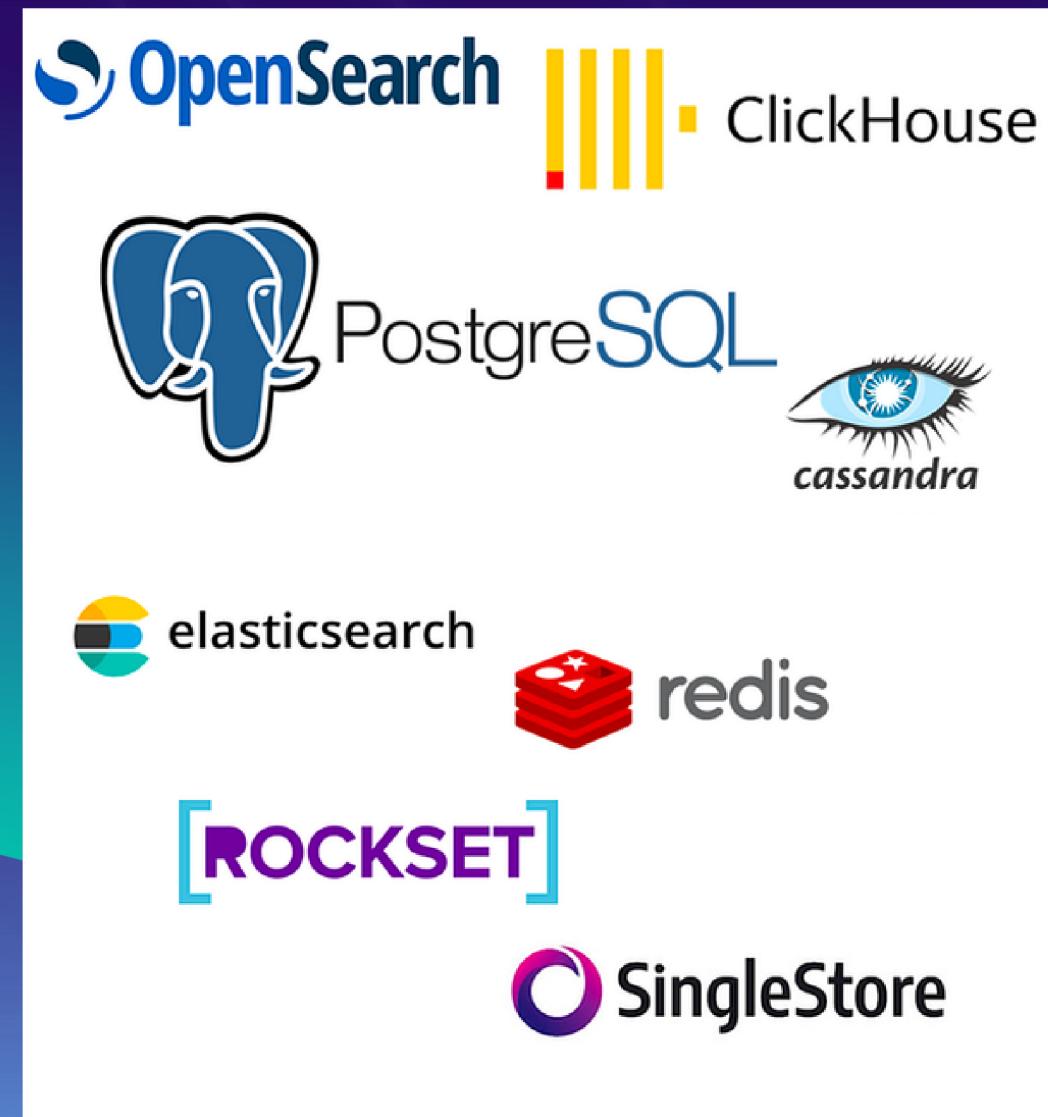
ANN  
Benchmarks



# [2] Pretraga vektorskih podataka

## Tradicionalne baze podataka sa podrškom za vektorsku pretragu

Tradicionalne baze podataka sa dodatkom za pretragu vektora predstavljaju postojeće baze podataka koje su se do sada koristile isključivo za pamćenje struktuiranih tipova podataka, bez uzimanja u obzir vektoru kao mogućeg tipa podatka.



[3]

# Vektorske baze podataka

---

Vektorske baze podataka

[3]

# [3] Vektorske baze podataka

---

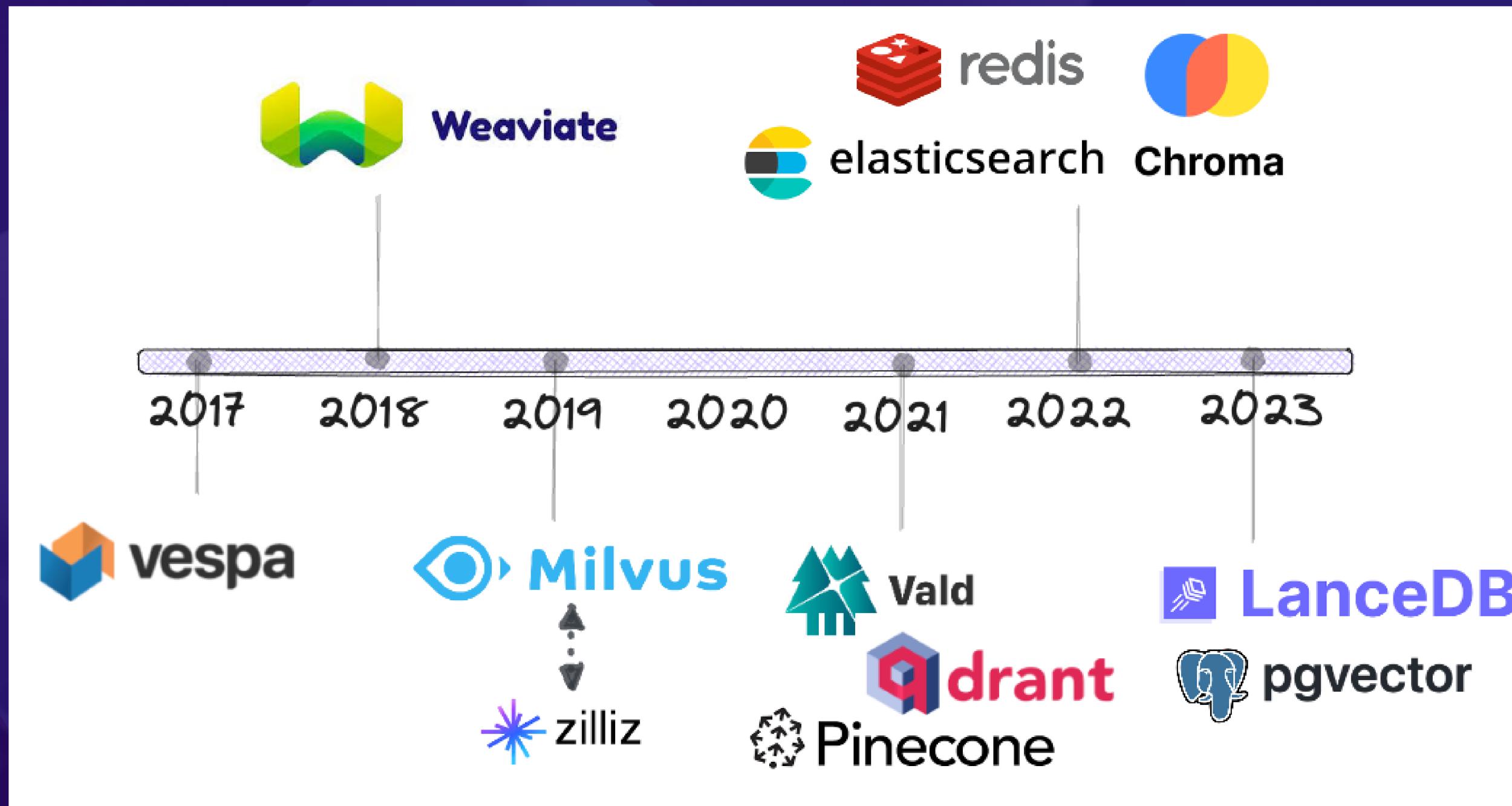
Vektorske baze podataka predstavljaju pristup prilagođen najnovijim trendovima u veštačkoj inteligenciji.

Specifično su kreirane kako bi mogle podržati rad nad velikim skupovima podataka i vršiti operacije nad velikim dimenzionalnostima vektora. Vektorske baze podržavaju veliki broj različitih tipova indeksiranja i metrika pretrage baze vektora, što čini dodatno poboljšanje u pogledu performansi.

Kompanije koje razvijaju vektorske baze podataka uporedo rade na razvoju *On-Premise* i *Cloud-Native* rešenja za svoje proizvode te se stoga mogu podeliti prema načinu na koji se vrši *hosting* samih baza podataka.

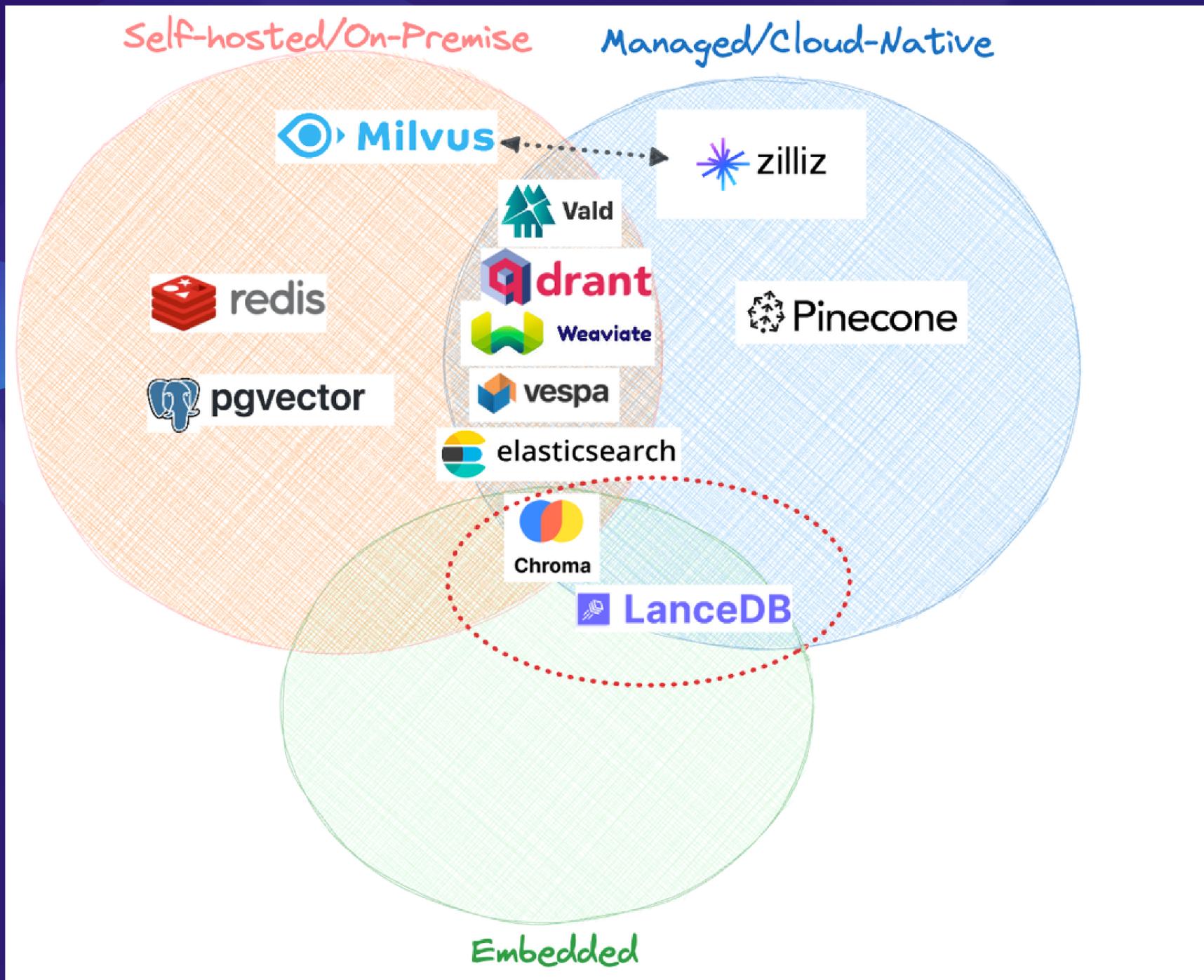
# [3] Vektorske baze podataka

## Vremenska linija nastanka



# [3] Vektorske baze podataka

## Podela prema Hosting-u



# [3] Vektorske baze podataka

## Zilliz + Milvus

*Open-source* vektorska baza podataka sa *On-Premise* i *Cloud-Native* rešenjem u obliku Zilliz platforme. Može se kreirati kao *Standalone* ili *Cluster* rešenje za skaliranje podataka.

Omogućava različite tipove indeksiranja kao što su *in-memory* i *disk-based* indeksiranja.

Algoritmi kod *in-memory* indeksiranja:

- FLAT
- IVF\_FLAT
- GRPU\_IVF\_FLAT
- IVF\_SQ8
- IVF\_PQ
- GPU\_IVF\_PQ
- HNSW
- SCANN

Algoritmi kod *disk-based* indeksiranja:

- DiskANN

# [3] Vektorske baze podataka

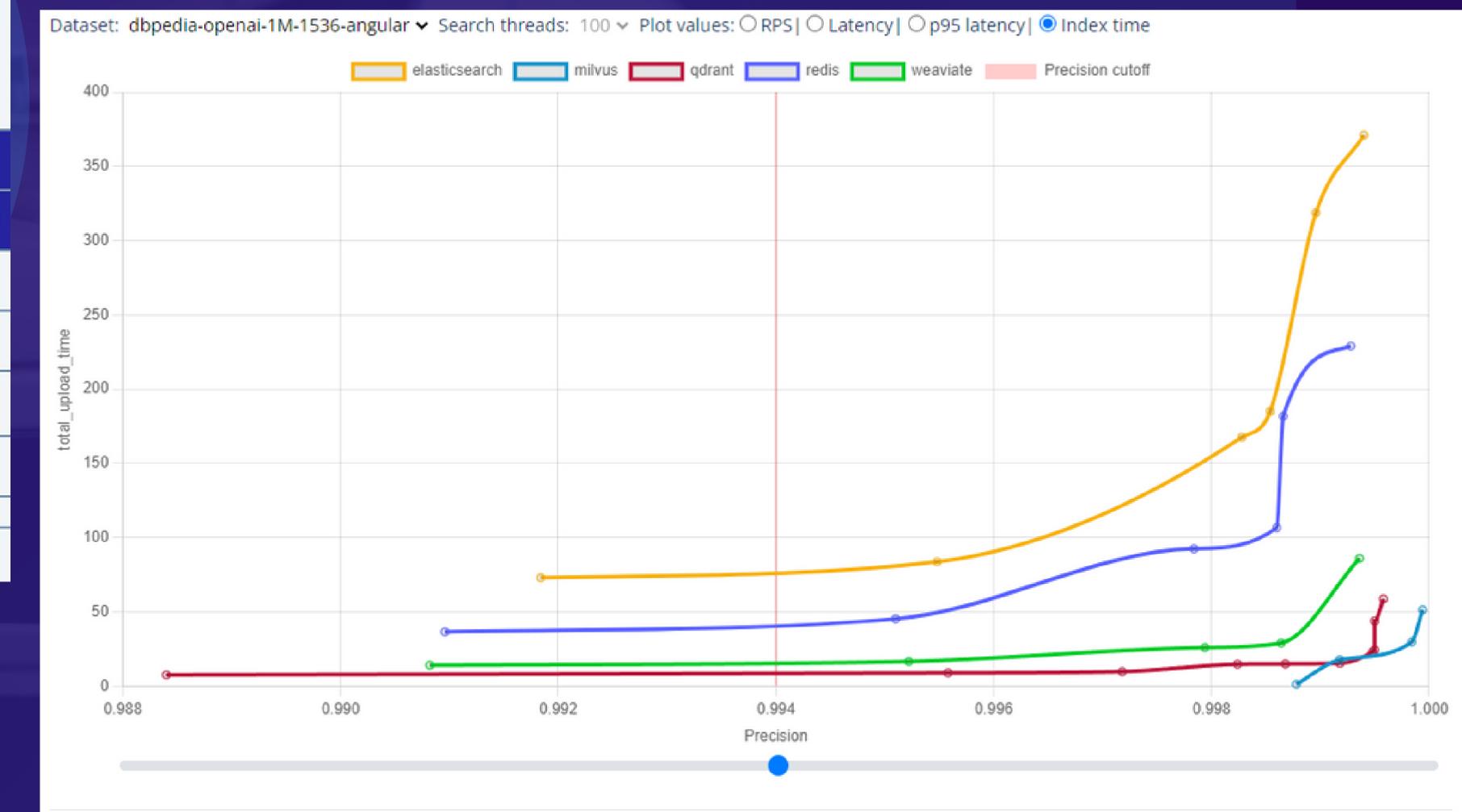
## NeurIPS'23 Competition Track: Big-ANN

Supported by Microsoft Pinecone AWS zilliz

New: the latest ongoing leaderboard has been released (March 1st, 2024).

Top entries:

Filter track			OOD track			Sparse track		
Rank	Algorithm	QPS@90% recall	Rank	Algorithm	QPS@90% recall	Rank	Algorithm	QPS@90% recall
1	Pinecone-filter	85,491	1	Pinecone-ood	38,088	1	Zilliz	10,749
2	Zilliz	84,596	2	Zilliz	33,241	2	Pinecone_smips	10,440
3	ParlayANN IVF <sup>2</sup>	37,902	3	RoarANN	22,555	3	PyANNS	8,732
4	Puck	19,193	4	PyANNS	22,296	4	shnsw	7,137
...	...	...	...	...	...	...	...	...
Baseline	FAISS	3,032	Baseline	Diskann	4,133	Baseline	Linscan	93



[4]

# Multimodalni modeli

---

Multimodalni modeli

[4]

## [4] Multimodalni modeli

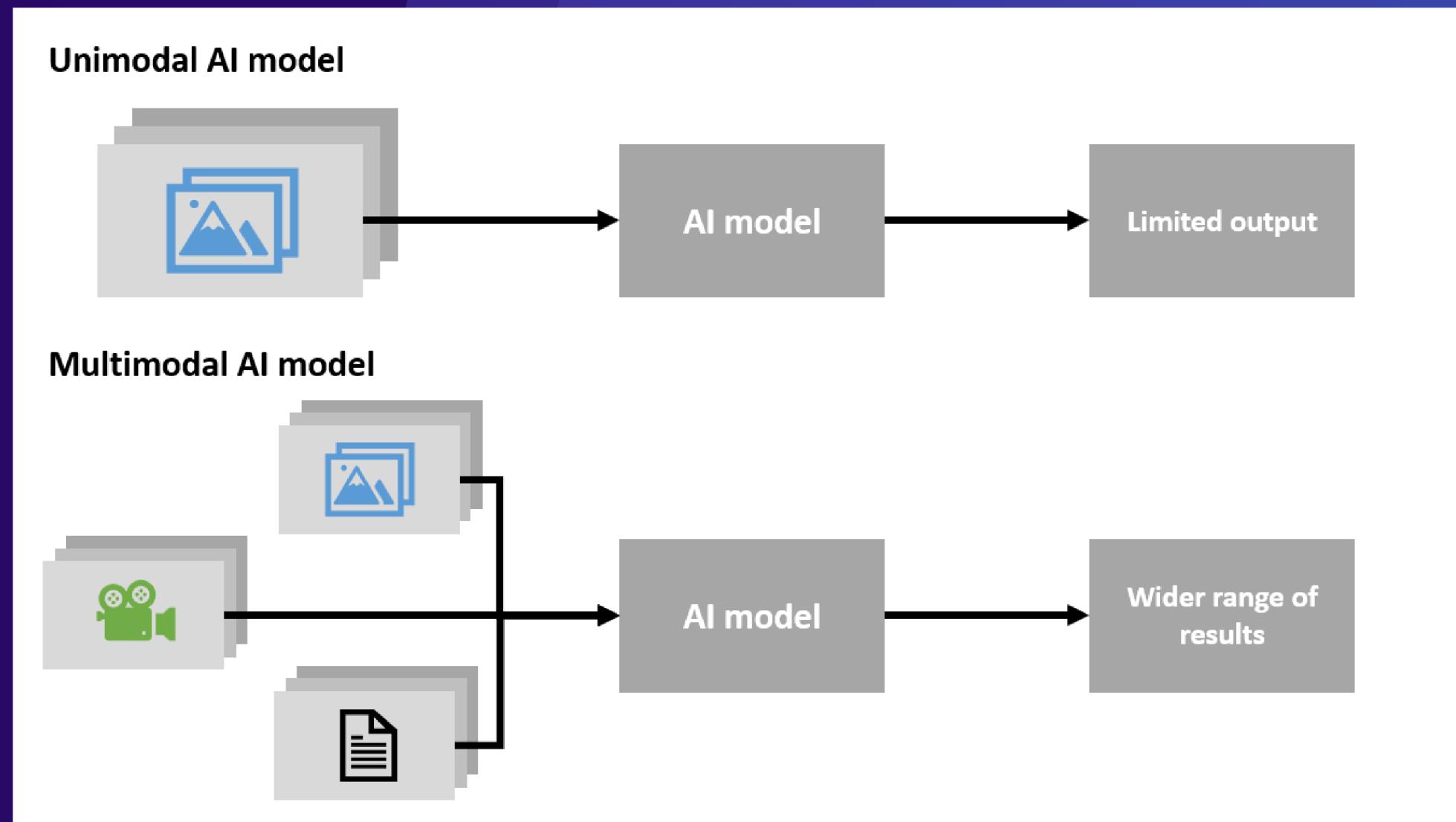
---

**Multimodalni modeli predstavljaju hibridni transformer model koji u sebi sadrži kombinaciju vizuelnog i tekstulnog enkodera izdvojenih iz transformer arhitekture.**

**Ovakvi hibridni modeli predstavljaju unapređenje dosadašnjih unimodalnih modela koji su imali sposobnost enkodiranja podataka isključivo jednog modaliteta u jedinstvenom vektorskom prostoru.**

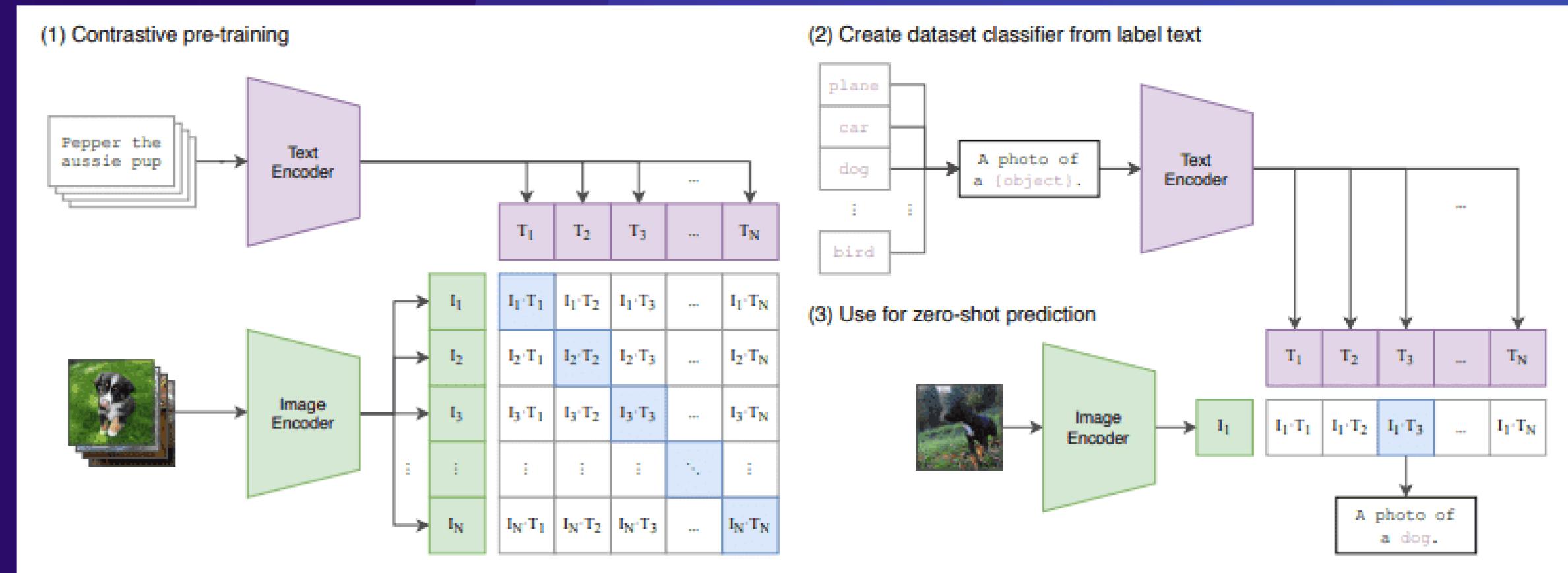
# [4] Multimodalni modeli

Enkodiranjem se višestruki modaliteti podataka poput zvuka, slike i teksta mogu predstaviti u identičnom vektorskom prostoru.



# [4] Multimodalni modeli

Treniranje ovakvog modela je bazirano na kontrastivnom učenju (*Eng. contrastive learning*) kod kojeg se u procesu treniranja prosleđuju parovi slika-tekst sa identičnim semantičkim značenjem. Na ovaj način, model treba utvrditi sličnost između identičnih parova.



# OpenCLIP

## Modeli kandidati:

- **xlm-roberta-large-ViT-H-14 , ver:  
frozen\_laion5b\_s13b\_b90k**
- **xlm-roberta-base-ViT-B-32 , ver:  
laion5b\_s13b\_b90k**

**OpenCLIP biblioteka predstavlja biblioteku razvijenu od strane open-source zajednice sa ciljem da se razvije skup multi-modalnih modela poput CLIP modela razvijenog od strane OpenAI kompanije. Ova biblioteka poseduje veliki broj pretreniranih multi-modalnih modela sa svim informacijama o tim modelima i skupovima podataka nad kojima su trenirani.**

[5]

# Aplikacija za vektorsku pretragu kućnih ljubimaca

---

Aplikacija za vektorsku  
pretragu kućnih ljubimaca

[5]

# Karakteristike aplikacije

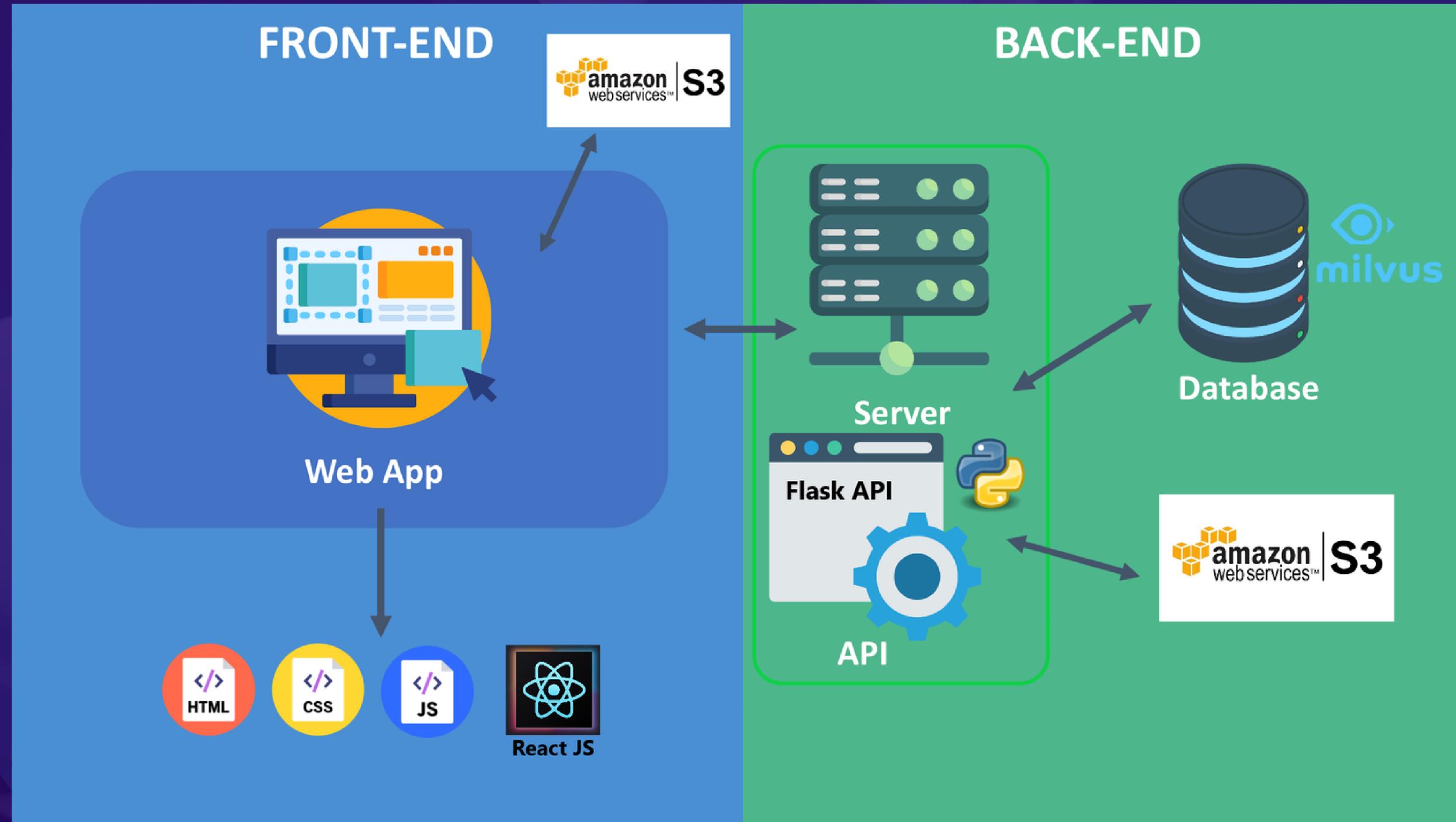
---

**Aplikacija omogućava pretraživanje pronađenih ili nestalih kućnih ljubimaca uz pomoć slike ili tekstualnog opisa.**

**Dodatna pretraga je omogućena navođenjem lokacije i radijusa oko navedene lokacije.**

- **Frontend aplikacije: React Framework**
- **Backend aplikacije: Flask Server**
- **AWS S3 servis za pamćenje slika**
- **Milvus vektorska baza podataka**
- **Hibridna pretraga**

# Arhitektura aplikacije



# Hibridna pretraga tekstom i slikom

**SEARCH MISSING PETS**

Describe your pet and search...

nemački ovčar sa braon šapama

**SEARCH BY DESCRIPTION**

Upload pet image and search...



3.jpg

Dog or cat

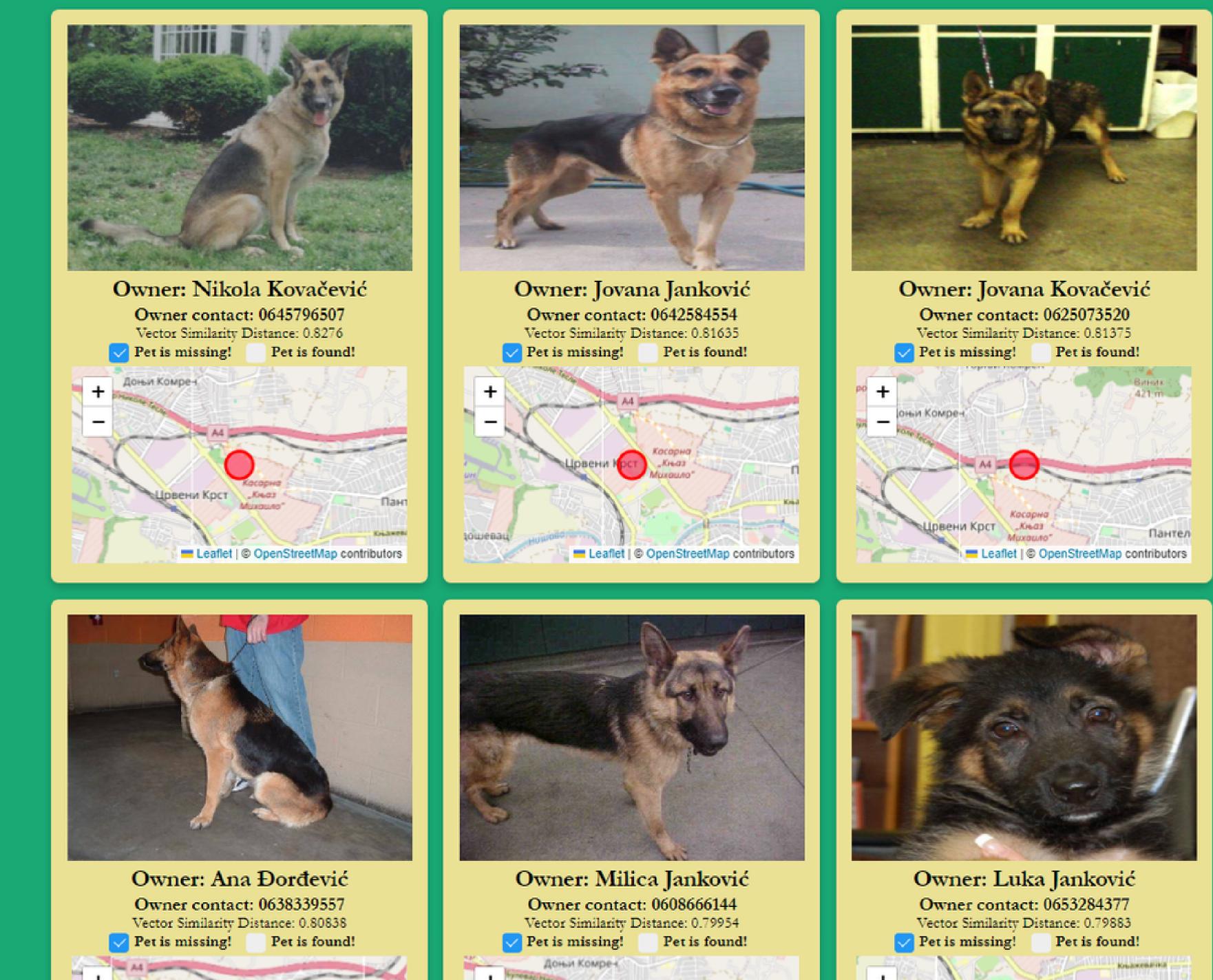
**SEARCH BY IMAGE**

Search pet with both image and pet description

Hybrid search is combination of visual and textual vector search

**HYBRID SEARCH**

Radius in meters:



The image displays a grid of nine search results for a German Shepherd dog. Each result card includes a photo of the dog, the owner's name, their contact information, a map showing the location of the find, and a checkbox indicating if the pet is missing or found.

Owner	Contact	Vector Similarity Distance	Status
Nikola Kovačević	0645796507	0.8276	<input checked="" type="checkbox"/> Pet is missing! <input type="checkbox"/> Pet is found!
Jovana Janković	0642584554	0.81635	<input checked="" type="checkbox"/> Pet is missing! <input type="checkbox"/> Pet is found!
Jovana Kovačević	0625073520	0.81375	<input checked="" type="checkbox"/> Pet is missing! <input type="checkbox"/> Pet is found!
Ana Đorđević	0638339557	0.80838	<input checked="" type="checkbox"/> Pet is missing! <input type="checkbox"/> Pet is found!
Milica Janković	0608666144	0.79954	<input checked="" type="checkbox"/> Pet is missing! <input type="checkbox"/> Pet is found!
Luka Janković	0653284377	0.79883	<input checked="" type="checkbox"/> Pet is missing! <input type="checkbox"/> Pet is found!
(Top Left)	(Top Middle)	(Top Right)	
(Bottom Left)	(Bottom Middle)	(Bottom Right)	

# Praktična demonstracija aplikacije

---

Praktična  
demonstracija aplikacije

# [6] Zaključak

---

Budući rad u ovoj oblasti može uključiti:

- Dodatne eksperimente sa novim arhitekturama modela
- *Finetuning* modela za specifične slučajeve korišćenja
- Primenu novih algoritama za indeksiranje vektora
- Proširivanje podataka tekstualnim informacijama o kućnim ljubimcima



# HVALA NA PAŽNJI

---

