



elastic

The Search
AI Company

BBL at

sopra  steria



```
$ curl -XPOST https://localhost:9200/speaker/_doc -d '{
  "name" : "David Pilato",
  "jobs" : [
    { "name" : "SRA Europe (SSII)", "date" : "1995" },
    { "name" : "SFR", "date" : "1997" },
    { "name" : "e-Brands / Vivendi", "date": "2000" },
    { "name" : "DGDDI (douane)", "date" : "2005" },
    { "name" : "elastic", "date" : "2013" }
  ],
  "motivations" : [ "family", "job", "deejay" ],
  "blog" : "https://david.pilato.fr/",
  "twitter" : [ "@dadoonet", "@elasticfr" ],
  "bluesky" : [ "@pilato.fr" ],
  "email" : "david@elastic.co"
}' -H 'Content-Type: application/json'
```

So Many New Features One Search AI Platform

Build Your Own



Elasticsearch

Out-of-the-Box Solutions



Observability



Security

Search AI Platform



Ingest



Process



Storage &
Replication



Search



AI & ML
Analysis



Visualization



Workflow
Automation

Search AI Lake

start-local

Elasticsearch — the most widely deployed vector database

Copy to try locally in two minutes

```
curl -fsSL https://elastic.co/start-local | sh
```



[Read docs](#) →

OR

Deploy for production

[Start free cloud trial](#)

Or, [download on-prem](#)

Elastic pricing

The best way to consume Elastic is Elastic Cloud, a public cloud managed service. Elastic Cloud is available on your preferred cloud provider — AWS, Azure, or Google Cloud. Customers who want to manage the software themselves, whether on public, private, or hybrid cloud, can download the Elastic Stack.

[Start free trial](#)[Estimate costs →](#)

Standard

As low as
\$95 per month¹

[Try free](#)

A great place to start

- ✓ Core Elastic Stack features, including security
- ✓ Discover, field statistics, Kibana Lens, Elastic Maps, and Canvas
- ✓ Alerting and in-stack action

Gold

As low as
\$109 per month¹

[Try free](#)

Everything in Standard plus:

- ✓ Reporting
- ✓ Third-party alerting actions
- ✓ Watcher
- ✓ Multi-stack monitoring

Platinum

As low as
\$125 per month¹

[Try free](#)

Everything in Gold plus:

- ✓ Advanced Elastic Stack security features
- ✓ Machine learning (ML) – anomaly detection, supervised learning, third-party model management
- ✓ Cross-cluster replication

Enterprise

As low as
\$175 per month¹

[Try free](#)

Everything in Platinum plus:

- ✓ Searchable snapshots
- ✓ Support for searchable cold and frozen tiers
- ✓ Elastic Maps Server

serverless

General purpose

Best for general search use cases across various data types.

[Try for free](#)

Vector search

Best for semantic search use cases using vectors with near-real-time retrieval.

[Try for free](#)

Time series

Best for retention and analysis of high volume time series data, such as logs of other data streams.

Coming soon

Ingest Ingest VCUs for indexing data	\$0.14 Per VCU-hour	\$0.14 Per VCU-hour	Coming soon
Search Search VCUs for querying data	\$0.09 Per VCU-hour	\$0.09 Per VCU-hour	Coming soon
Machine Learning Machine Learning VCUs for trained models and ML jobs	\$0.07 Per VCU-hour	\$0.07 Per VCU-hour	Coming soon
Storage Persistent storage in Search AI Lake	\$0.047 Per GB-month	\$0.047 Per GB-month	Coming soon
Egress Data transfer out of project 50 GB monthly allowance per organization	\$0.05 Per GB	\$0.05 Per GB	Coming soon

A typical search implementation...

```
CREATE TABLE user
(
  name VARCHAR(100),
  comments VARCHAR(1000)
);
INSERT INTO user VALUES ('David Pilato', 'Developer at elastic');
INSERT INTO user VALUES ('Malloum Laya', 'Worked with David at french
customs service');
INSERT INTO user VALUES ('David Gageot', 'Engineer at Doctolib');
INSERT INTO user VALUES ('David David', 'Who is that guy?');
```



Search on term

```
INSERT INTO user VALUES ('David Pilato', 'Developer at elastic');
INSERT INTO user VALUES ('Malloum Laya', 'Worked with David at french
customs service');
INSERT INTO user VALUES ('David Gageot', 'Engineer at Doctolib');
INSERT INTO user VALUES ('David David', 'Who is that guy?');
```

```
SELECT * FROM user WHERE name="David";
Empty set (0,00 sec)
```



Search like

```
INSERT INTO user VALUES ('David Pilato', 'Developer at elastic');
INSERT INTO user VALUES ('Malloum Laya', 'Worked with David at french
customs service');
INSERT INTO user VALUES ('David Gageot', 'Engineer at Doctolib');
INSERT INTO user VALUES ('David David', 'Who is that guy?');
```

```
SELECT * FROM user WHERE name LIKE "%David%";
```

name	comments
David Pilato	Developer at elastic
David Gageot	Engineer at Doctolib
David David	Who is that guy?



Search for terms

```
INSERT INTO user VALUES ('David Pilato', 'Developer at elastic');
INSERT INTO user VALUES ('Malloum Laya', 'Worked with David at french
customs service');
INSERT INTO user VALUES ('David Gageot', 'Engineer at Doctolib');
INSERT INTO user VALUES ('David David', 'Who is that guy?');
```

```
SELECT * FROM user WHERE name LIKE "%David Pilato%";
```

name	comments
David Pilato	Developer at elastic



Search with inverted terms

```
INSERT INTO user VALUES ('David Pilato', 'Developer at elastic');
INSERT INTO user VALUES ('Malloum Laya', 'Worked with David at french
customs service');
INSERT INTO user VALUES ('David Gageot', 'Engineer at Doctolib');
INSERT INTO user VALUES ('David David', 'Who is that guy?');
```

```
SELECT * FROM user WHERE name LIKE "%Pilato David%";
```

Empty set (0,00 sec)

```
SELECT * FROM user WHERE name LIKE "%Pilato%David%";
```

Empty set (0,00 sec)



Search for terms

```
INSERT INTO user VALUES ('David Pilato', 'Developer at elastic');
INSERT INTO user VALUES ('Malloum Laya', 'Worked with David at french
customs service');
INSERT INTO user VALUES ('David Gageot', 'Engineer at Doctolib');
INSERT INTO user VALUES ('David David', 'Who is that guy?');
```

```
SELECT * FROM user WHERE name LIKE "%David%" AND
                        name LIKE "%Pilato%";
```

name	comments
David Pilato	Developer at elastic



Search in two fields

```
INSERT INTO user VALUES ('David Pilato', 'Developer at elastic');
INSERT INTO user VALUES ('Malloum Laya', 'Worked with David at french
customs service');
INSERT INTO user VALUES ('David Gageot', 'Engineer at Doctolib');
INSERT INTO user VALUES ('David David', 'Who is that guy?');
```

```
SELECT * FROM user WHERE name LIKE "%David%" OR
        comments LIKE "%David%";
```

name	comments
David Pilato	Developer at elastic
Malloum Laya	Worked with David at french customs service
David Gageot	Engineer at Doctolib
David David	Who is that guy?





Search with typos

```
INSERT INTO user VALUES ('David Pilato', 'Developer at elastic');
INSERT INTO user VALUES ('Malloum Laya', 'Worked with David at french
customs service');
INSERT INTO user VALUES ('David Gageot', 'Engineer at Doctolib');
INSERT INTO user VALUES ('David David', 'Who is that guy?');
```

```
SELECT * FROM user WHERE name LIKE "%Dadid%";
Empty set (0,00 sec)
```



Search with typos

```
INSERT INTO user VALUES ('David Pilato', 'Developer at elastic');
INSERT INTO user VALUES ('Malloum Laya', 'Worked with David at french
customs service');
INSERT INTO user VALUES ('David Gageot', 'Engineer at Doctolib');
INSERT INTO user VALUES ('David David', 'Who is that guy?');
```

```
SELECT * FROM user WHERE name LIKE "%_adid%" OR
name LIKE "%D_did%" OR
name LIKE "%Da_id%" OR
name LIKE "%Dad_d%" OR
name LIKE "%Dadi_%";
```

name	comments
David Pilato	Developer at elastic
David Gageot	Engineer at Doctolib
David David	Who is that guy?





User Interface

Power Search:

ID Number	<input type="text"/>
Web Title	<input type="text"/>
Url	<input type="text"/>
Category	Select
Web Description	<input type="text"/>
Keywords	<input type="text"/>
Contact Name	<input type="text"/>
Contact Email	<input type="text"/>
Featured Links 🍌	Select ▾
Cool Links 🌟	Select ▾
Bold Links	Select ▾
Icon	<input type="radio"/> ⚠️ <input type="radio"/> 😄 <input type="radio"/> 📄 <input type="radio"/> 📄 <input type="radio"/> ✍️ <input type="radio"/> 🗣️
Rating Average ★★★★★	Select ▾
Number of Votes	between <input type="text"/> and <input type="text"/>
Total Hits	between <input type="text"/> and <input type="text"/>
Hits Today	between <input type="text"/> and <input type="text"/>
IP Address	<input type="text"/>
Submission Software Name	<input type="text"/>



Demo time!



Elasticsearch

You Know, for Search

```
GET /_analyze
{
  "char_filter": [ "html_strip" ],
  "tokenizer": "standard",
  "filter": [ "lowercase", "stop", "snowball" ],
  "text": "These are <em>not</em> the droids
          you are looking for."
}
```

```
"char_filter": "html_strip"
```

These are `not` the droids you are looking for.



These are not the droids you are looking for.

```
"tokenizer": "standard"
```

These are not the droids you are looking for.



```
These  
are  
not  
the  
droids  
you  
are  
looking  
for
```



```
"filter": "lowercase"
```

T hese	→	t hese
are		are
not		not
the		the
droids		droids
you		you
are		are
looking		looking
for		for

"filter": "stop"

T hese		t hese		
are		are		
not		not		
the		the		
droids	→	droids	→	droids
you		you		you
are		are		
looking		looking		looking
for		for		

"filter": "snowball"

These

are

not

the

droids



these

are

not

the

droids



droids



droid

you

are

looking

for

you

are

looking

for

you

look**ing**

you

look

These are `not` the **droids you** are **looking** for.

```
{ "tokens": [{
  "token": "droid",
  "start_offset": 27, "end_offset": 33,
  "type": "<ALPHANUM>", "position": 4
}, {
  "token": "you",
  "start_offset": 34, "end_offset": 37,
  "type": "<ALPHANUM>", "position": 5
}, {
  "token": "look",
  "start_offset": 42, "end_offset": 49,
  "type": "<ALPHANUM>", "position": 7
}]}
```



Elasticsearch

You Know, for **Vector** Search

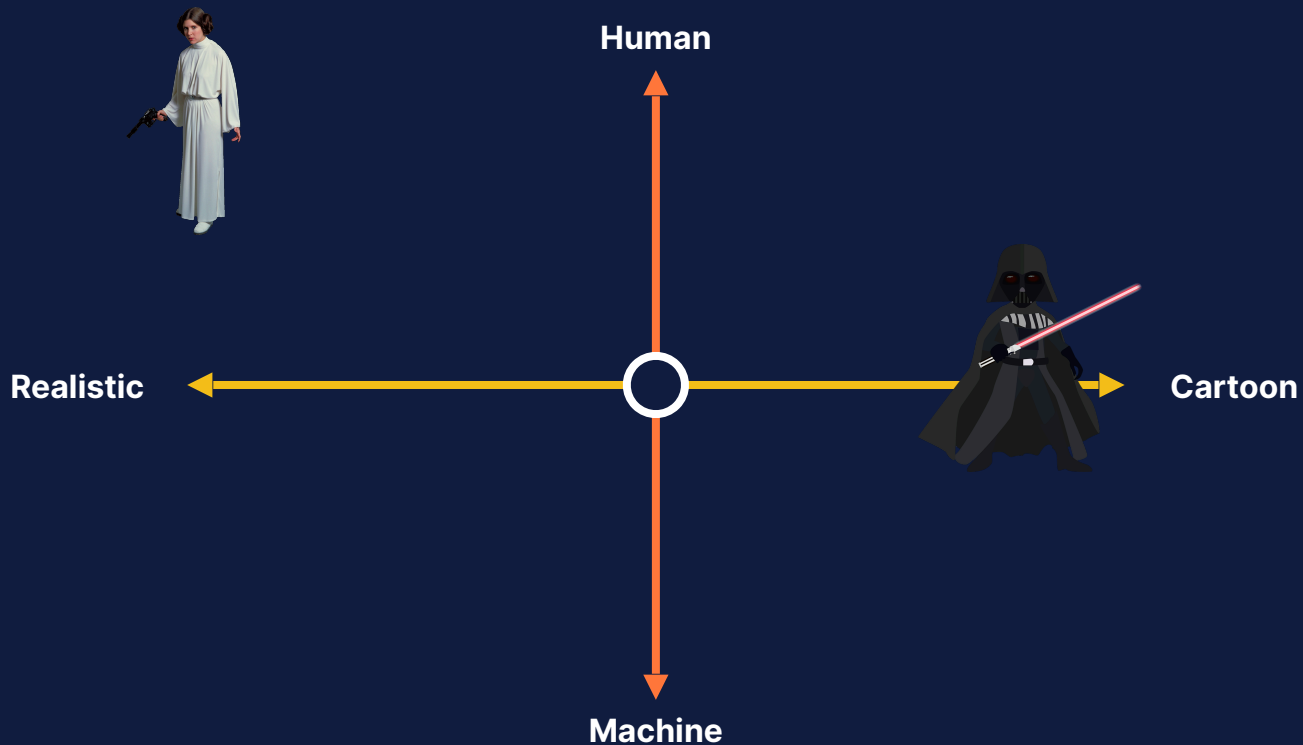
Embeddings represent your data



Example: 1-dimensional vector



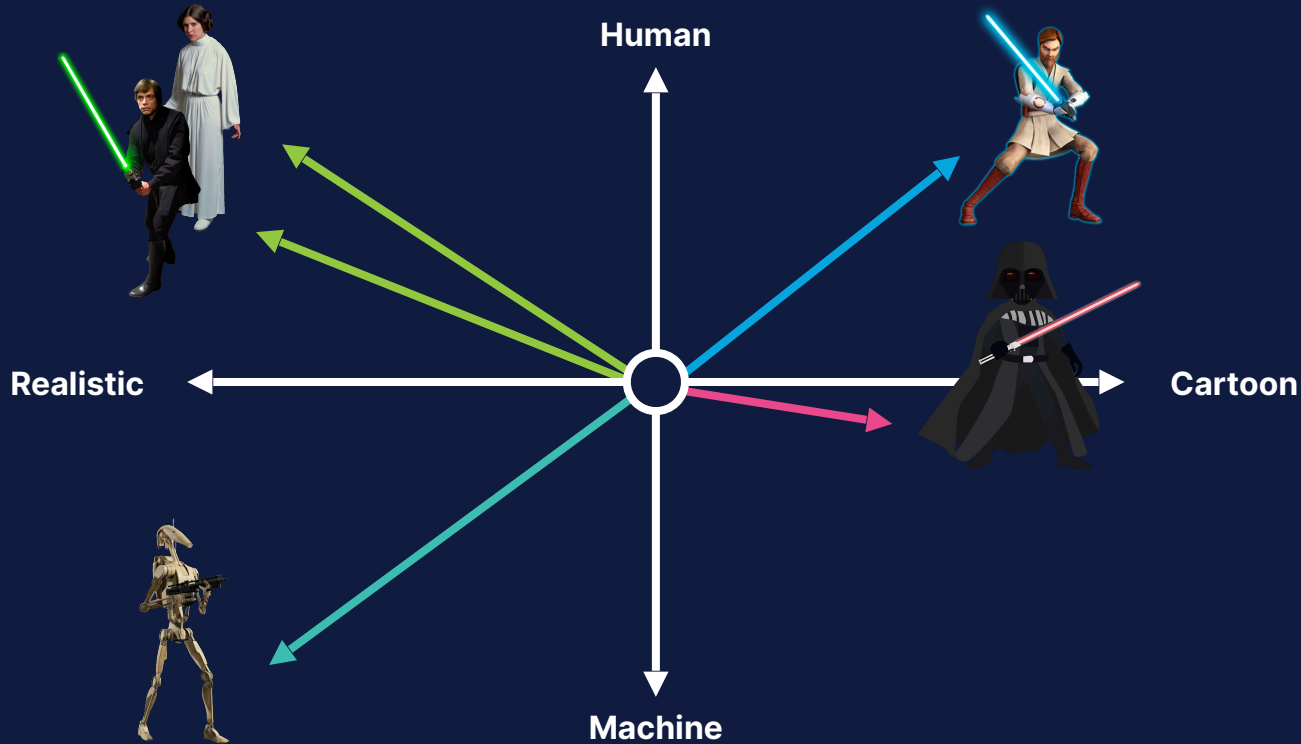
Character	Vector
	$[-1]$
	$[1]$




Multiple dimensions represent different data aspects



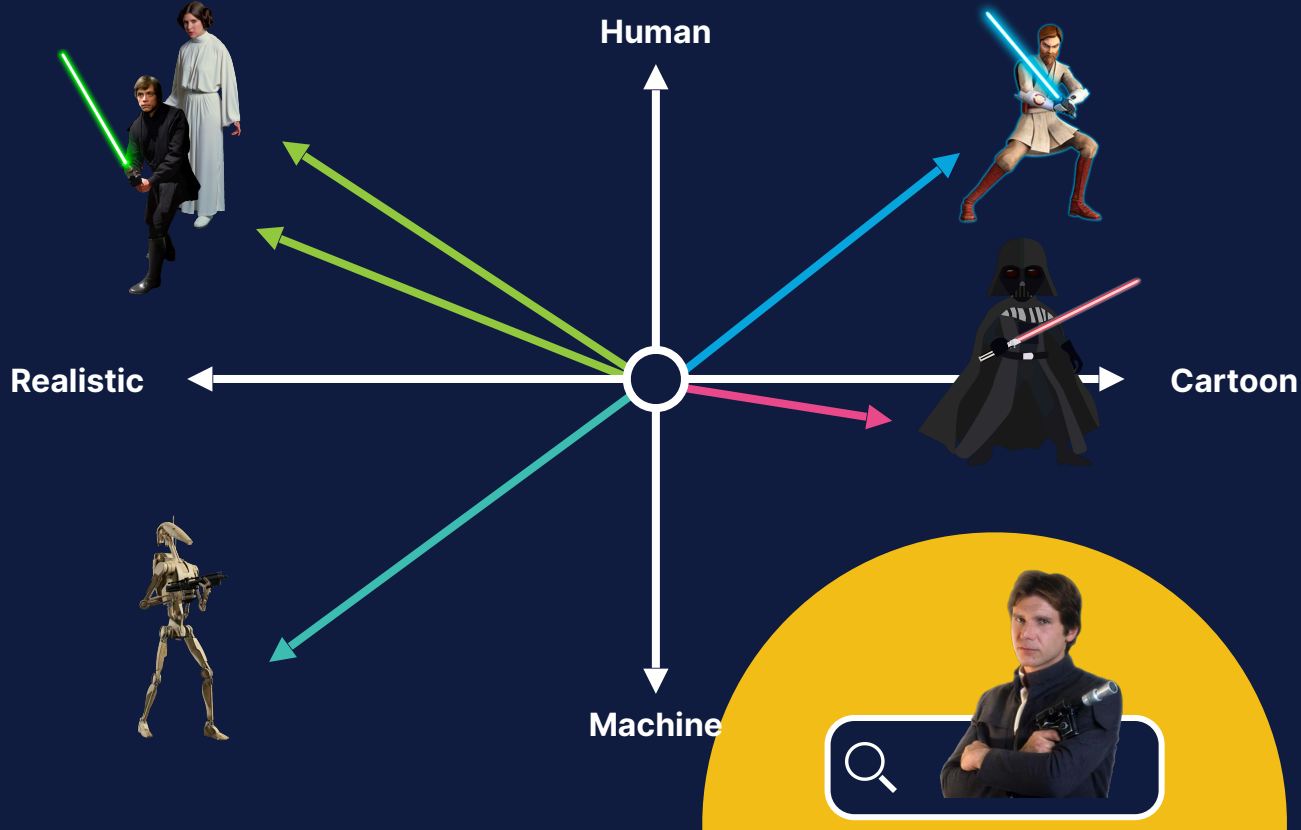
Character	Vector
	$[-1, 1]$
	$[1, 0]$

Similar data is grouped together



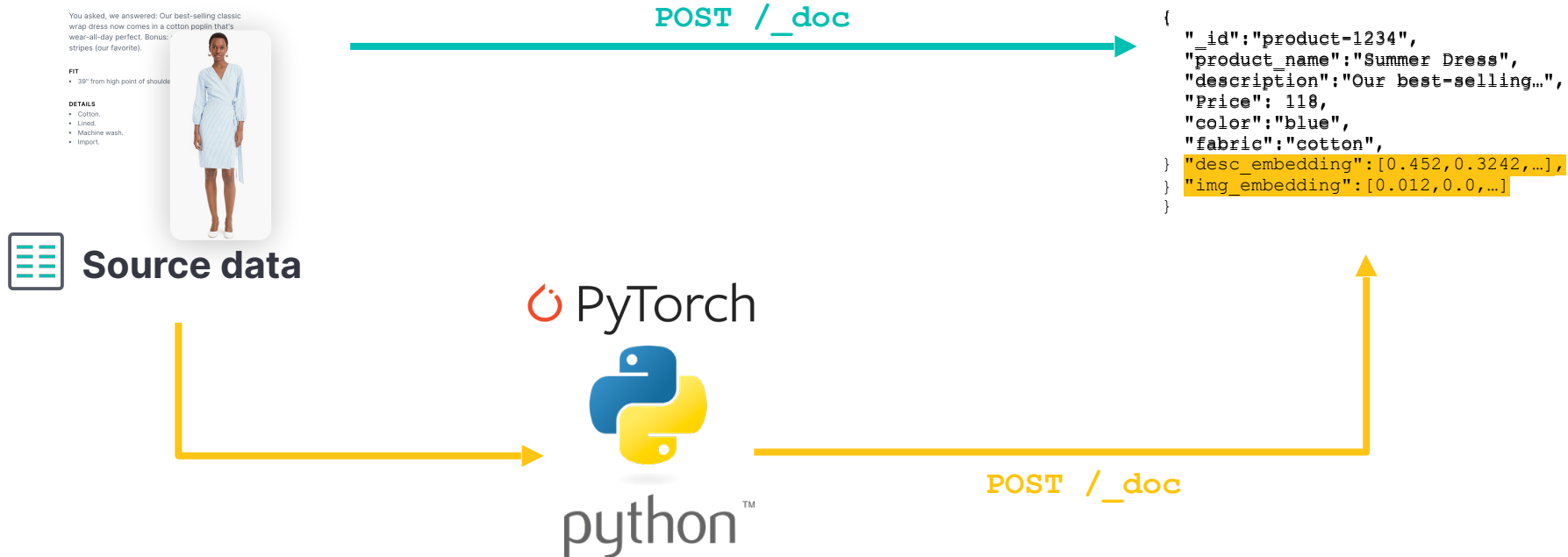
Character	Vector
	$[-1.0, 1.0]$
	$[1.0, 0.0]$
	$[-1.0, 0.8]$

Vector search ranks objects by similarity (~relevance) to the query



Rank	Result
Query	
1	
2	
3	
4	
5	

Data Ingestion and Embedding Generation



Vector Query

🔍 summer clothes ✕ 

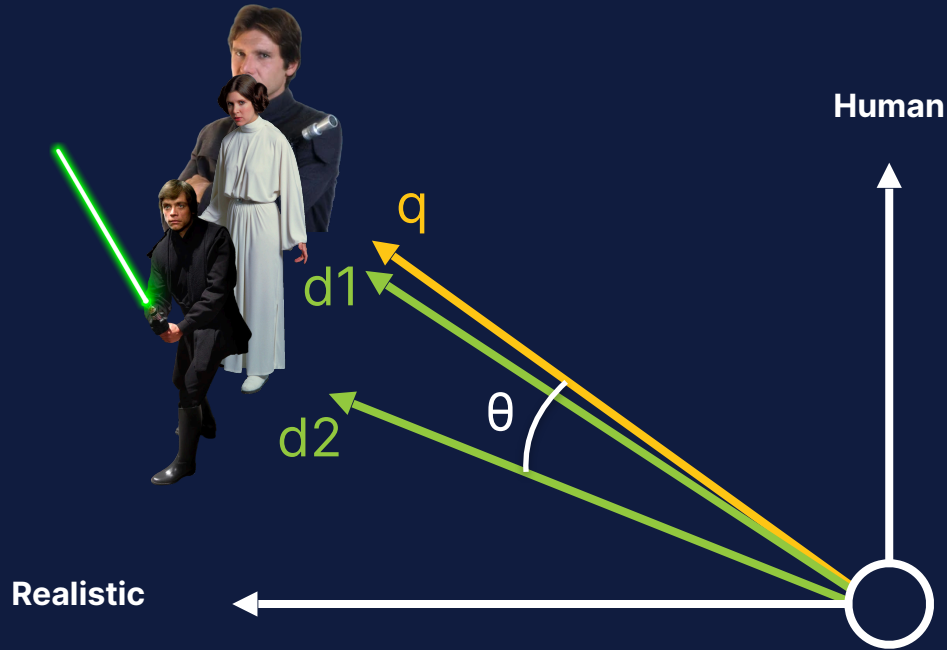
 PyTorch



python™

```
GET product-catalog/_search
{
  "query" : {
    "bool" : {
      "must" : [{
        "knn" : {
          "field": "desc_embedding",
          "num_candidates": 50,
          "query_vector": [0.123, 0.244, ...]
        }
      ]
    }
  },
  "filter" : {
    "term" : {
      "department": "women"
    }
  }
},
"size": 10
}
```

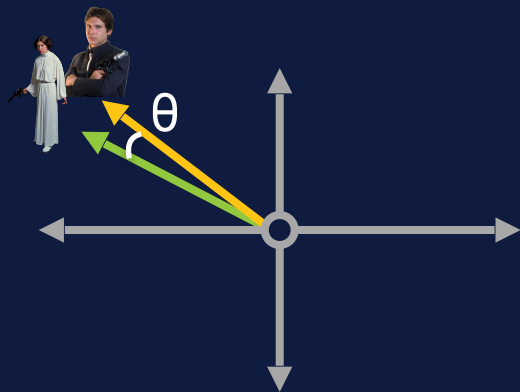
Similarity



$$\cos(\theta) = \frac{\vec{q} \cdot \vec{d}}{|\vec{q}| \times |\vec{d}|}$$

$$\text{_score} = \frac{1 + \cos(\theta)}{2}$$

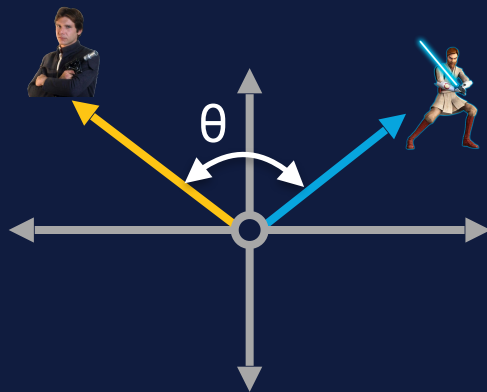
Similarity: cosine (cosine)



Similar vectors

θ close to 0
 $\cos(\theta)$ close to **1**

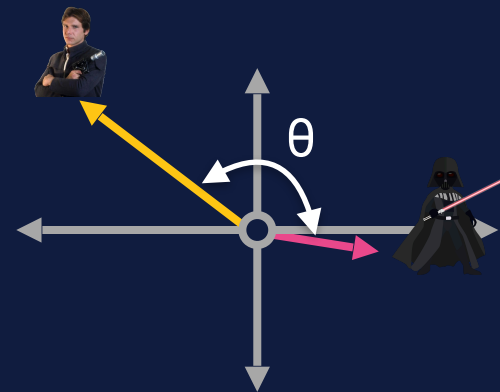
$$\text{_score} = \frac{1 + 1}{2} = 1$$



Orthogonal vectors

θ close to 90°
 $\cos(\theta)$ close to **0**

$$\text{_score} = \frac{1 + 0}{2} = 0.5$$



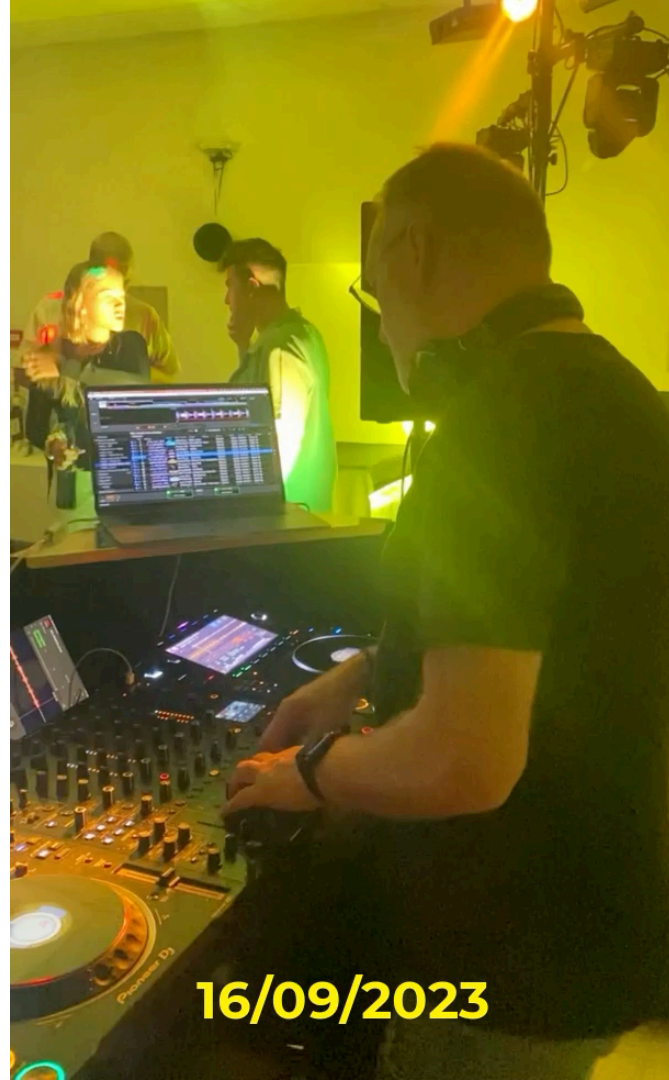
Opposite vectors

θ close to 180°
 $\cos(\theta)$ close to **-1**

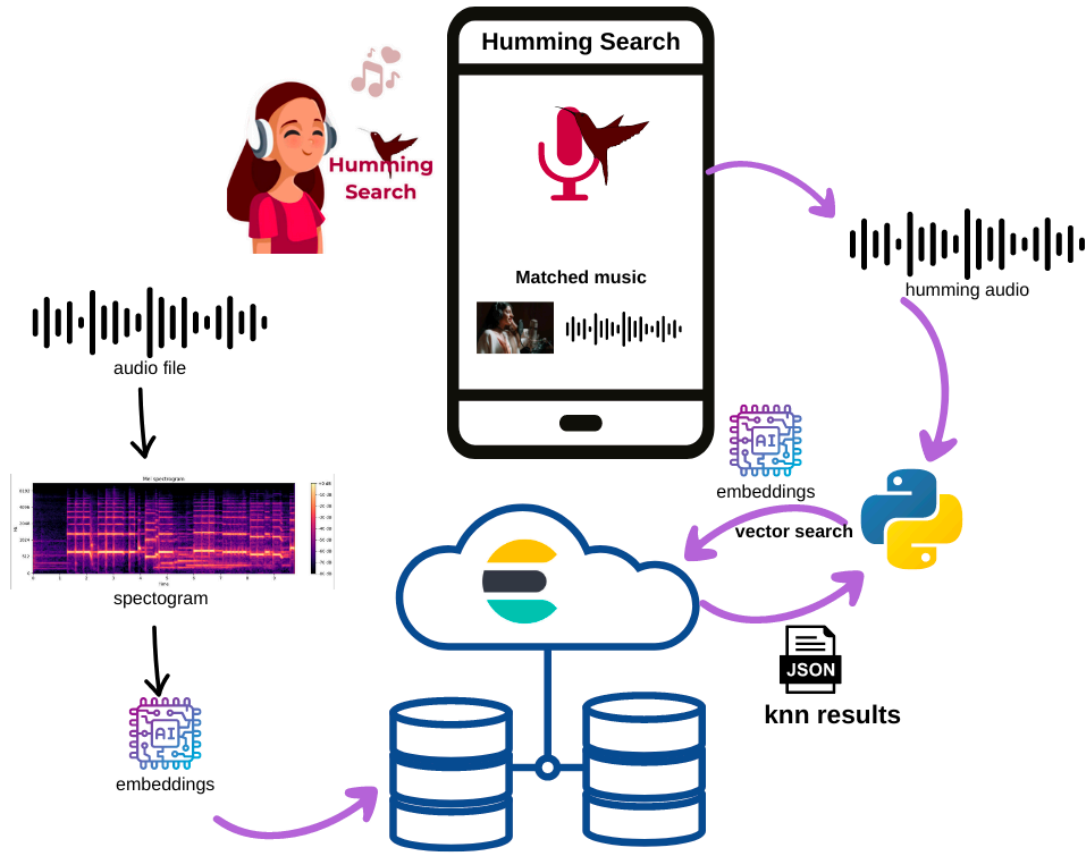
$$\text{_score} = \frac{1 - 1}{2} = 0$$



<https://djdadoo.pilato.fr/>



16/09/2023



<https://github.com/dadoonet/music-search/>

So Many New Features One **Search AI** Platform

Build Your Own



Elasticsearch

Out-of-the-Box Solutions



Observability



Security

Search AI Platform



Ingest



Process



Storage &
Replication



Search



AI & ML
Analysis



Visualization



Workflow
Automation

Search AI Lake



www.meetup.com/ElasticFR



@elasticfr



discuss.elastic.co



Thank You

