

 2023

SPRING  10^{TH ANNIVERSARY}

#springio23

Vector Similarity Search

in Spring with **Redis** Stack



Brian Sam-Bodden



BARCELONA MAY 18-19 / WWW.SPRINGIO.NET



bsb@redis.com



[@bsbodden](https://twitter.com/bsbodden)



github.com/bsbodden



Vector Similarity Search

TLDR

Why? Vast majority of data is **Unstructured** data!

What? **Vector Databases** store vectors efficiently

How? **Redis** provides **Vector Storage** and **Search Capabilities**

Vector Similarity Search

TLDR

Why? Vast majority of data is **Unstructured** data!



What? **Vector Databases** store vectors efficiently

How? **Redis** provides **Vector Storage** and **Search Capabilities**

Vector Similarity Search

TLDR

Why? Vast majority of data is **Unstructured** data!

What? **Vector Databases** store vectors efficiently

How? **Redis** provides **Vector Storage** and **Search Capabilities**

Vector Similarity Search

TLDR

Why? Vast majority of data is **Unstructured** data!

What? **Vector Databases** store vectors efficiently

How? **Redis** provides **Vector Storage** and **Search Capabilities**



REmote DI_ctio_nary Server

REmote DI~~c~~tionary Server

REmote Dictionary Server

REmote Dictionary Server



In-memory First



Optionally Persistent



In-memory First



Optionally Persistent



OSS



OSS



1



1

Strings



OSS

1

- Sets**
- Strings**



1

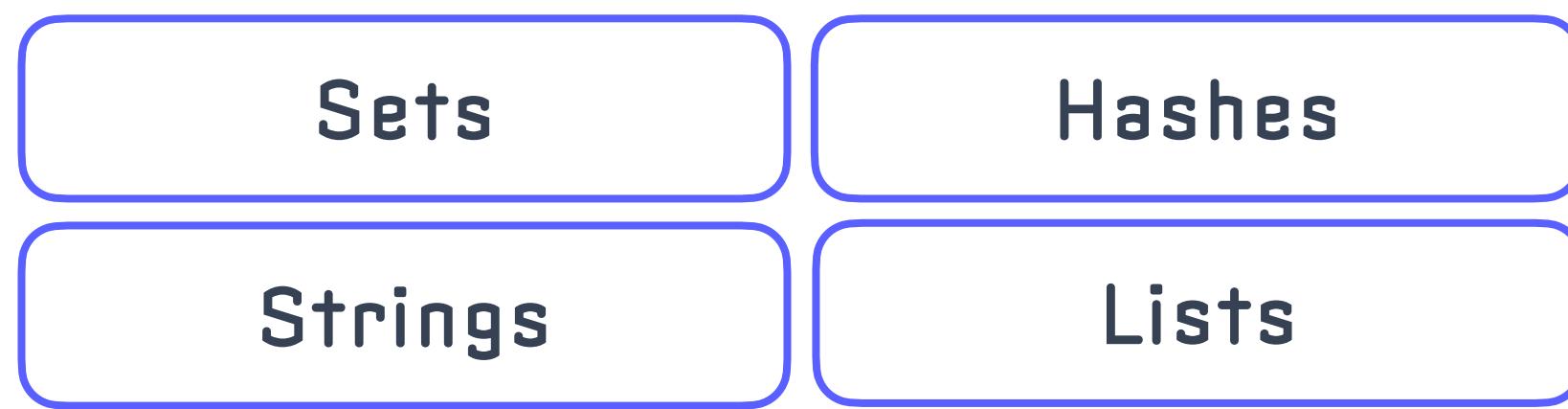
Sets

Strings

Lists



1





OSS

1

Sets

Hashes

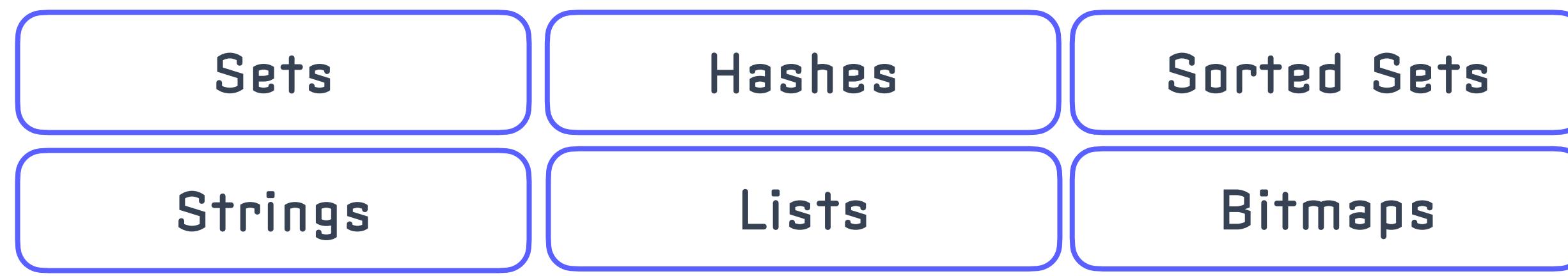
Strings

Lists

Bitmaps

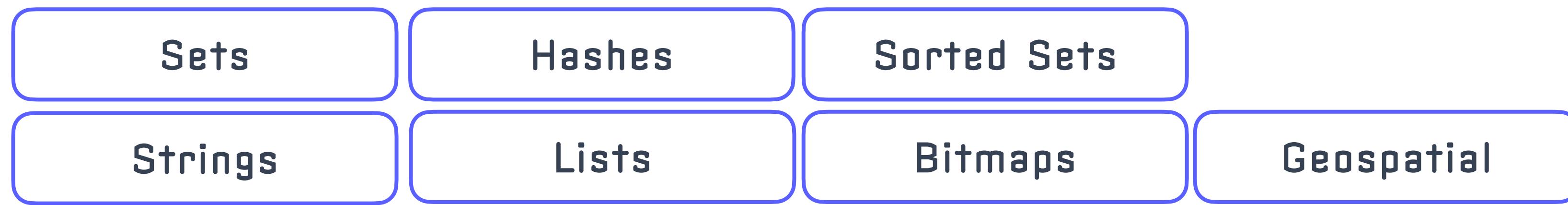


1





1





1

Sets

Hashes

Sorted Sets

Bit Field

Strings

Lists

Bitmaps

Geospatial



OSS

1

Sets

Hashes

Sorted Sets

Bit Field

Strings

Lists

Bitmaps

Geospatial

Streams



OSS

1

Sets

Hashes

Sorted Sets

Bit Field

HyperLogLog

Strings

Lists

Bitmaps

Geospatial

Streams



OSS

1

Sets

Hashes

Sorted Sets

Bit Field

HyperLogLog

Strings

Lists

Bitmaps

Geospatial

Streams

Foundational Data Structures



1

Sets

Hashes

Sorted Sets

Bit Field

HyperLogLog

Strings

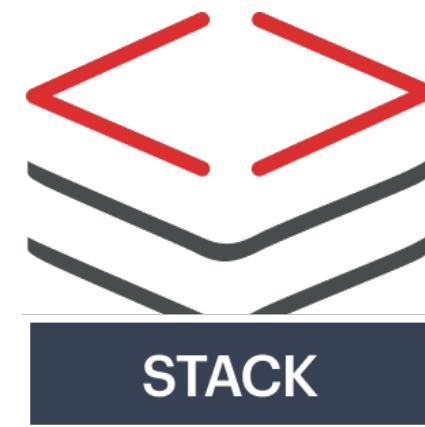
Lists

Bitmaps

Geospatial

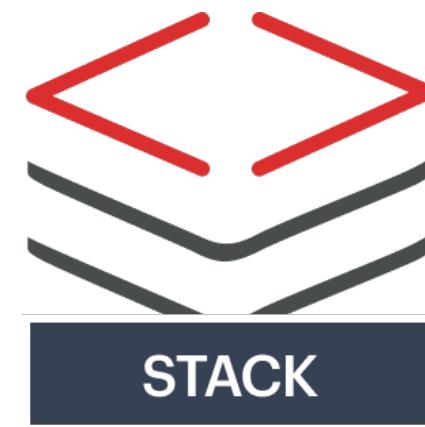
Streams

Foundational Data Structures



1

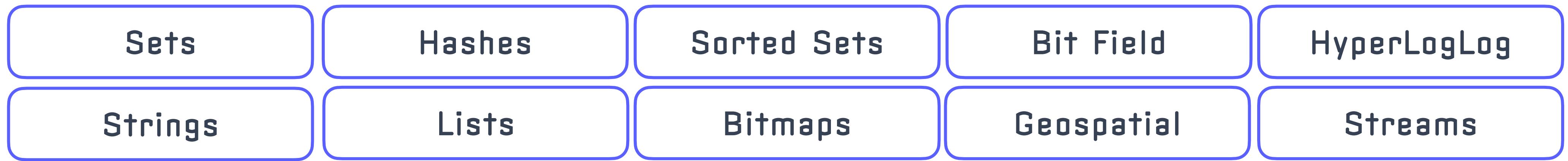
Sets**Hashes****Sorted Sets****Bit Field****HyperLogLog****Strings****Lists****Bitmaps****Geospatial****Streams****Foundational Data Structures**

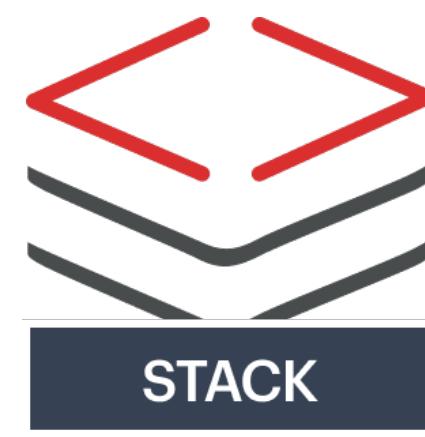


2



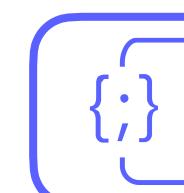
1

**Foundational Data Structures**



STACK

2



JSON



OSS

1

Sets

Hashes

Sorted Sets

Bit Field

HyperLogLog

Strings

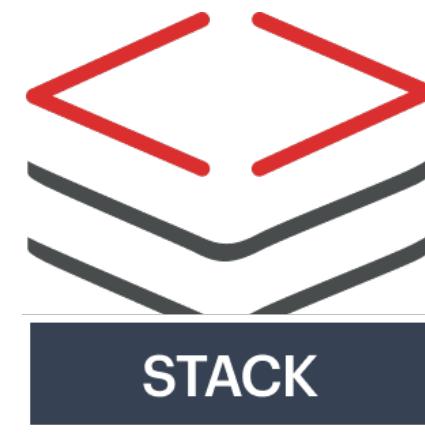
Lists

Bitmaps

Geospatial

Streams

Foundational Data Structures



2



JSON



Probabilistic



1

Sets

Hashes

Sorted Sets

Bit Field

HyperLogLog

Strings

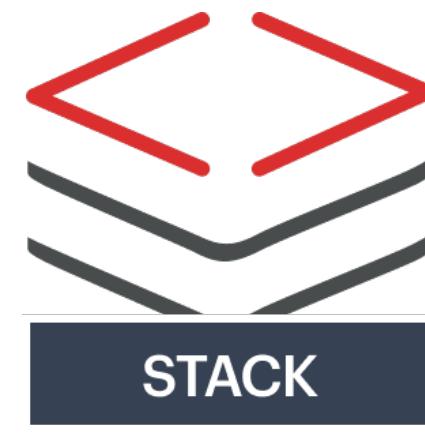
Lists

Bitmaps

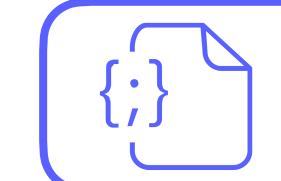
Geospatial

Streams

Foundational Data Structures



2



JSON



Probabilistic

Documents and Probabilistic Data Structures



1

Sets

Hashes

Sorted Sets

Bit Field

HyperLogLog

Strings

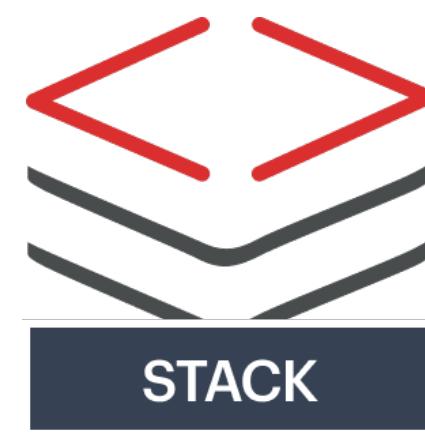
Lists

Bitmaps

Geospatial

Streams

Foundational Data Structures



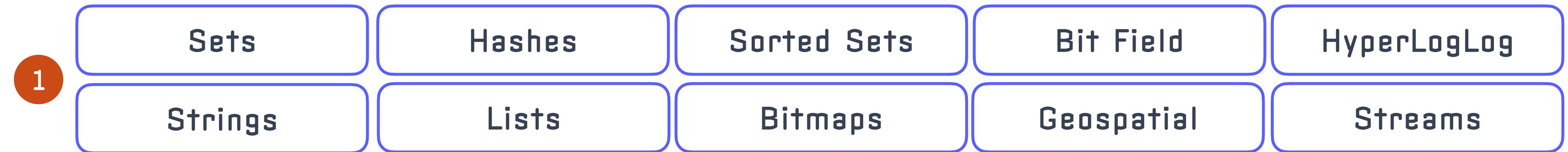
3

2

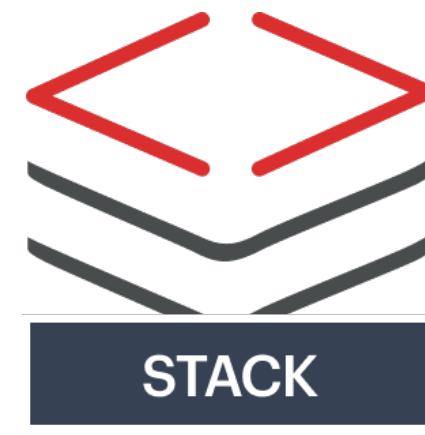
1



Documents and Probabilistic Data Structures



Foundational Data Structures



3

**Search**

2

**JSON****Probabilistic**

Documents and Probabilistic Data Structures



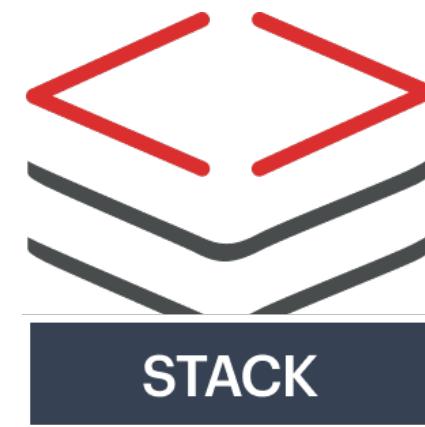
1

Sets**Hashes****Sorted Sets****Bit Field****HyperLogLog**

OSS

Strings**Lists****Bitmaps****Geospatial****Streams**

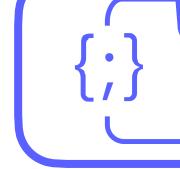
Foundational Data Structures



3

**Search****Functions**

2

**JSON****Probabilistic**

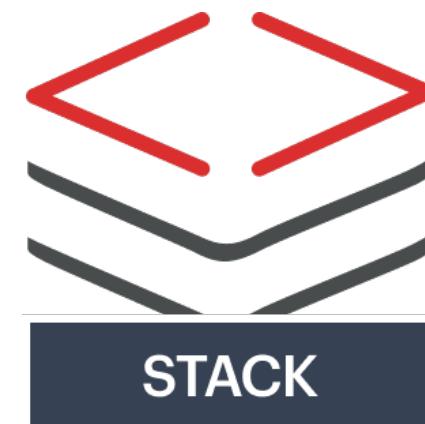
Documents and Probabilistic Data Structures



1

Sets**Hashes****Sorted Sets****Bit Field****HyperLogLog****Strings****Lists****Bitmaps****Geospatial****Streams**

Foundational Data Structures



STACK



OSS

3



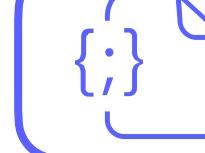
Search



Functions

Redis-side Computing

2



JSON



Probabilistic

Documents and Probabilistic Data Structures

1

Sets

Hashes

Sorted Sets

Bit Field

HyperLogLog

Strings

Lists

Bitmaps

Geospatial

Streams

Foundational Data Structures

4

3



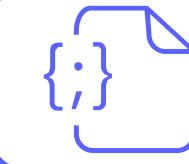
Search



Functions

Redis-side Computing

2



JSON



Probabilistic

Documents and Probabilistic Data Structures

1

Sets

Hashes

Sorted Sets

Bit Field

HyperLogLog

Strings

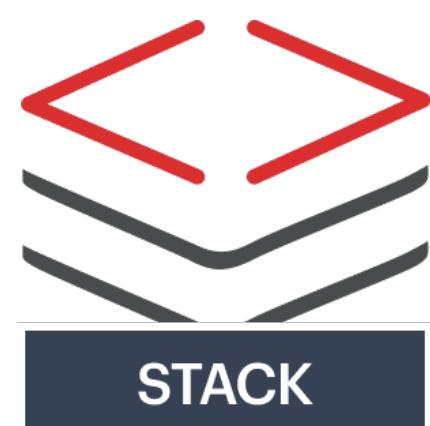
Lists

Bitmaps

Geospatial

Streams

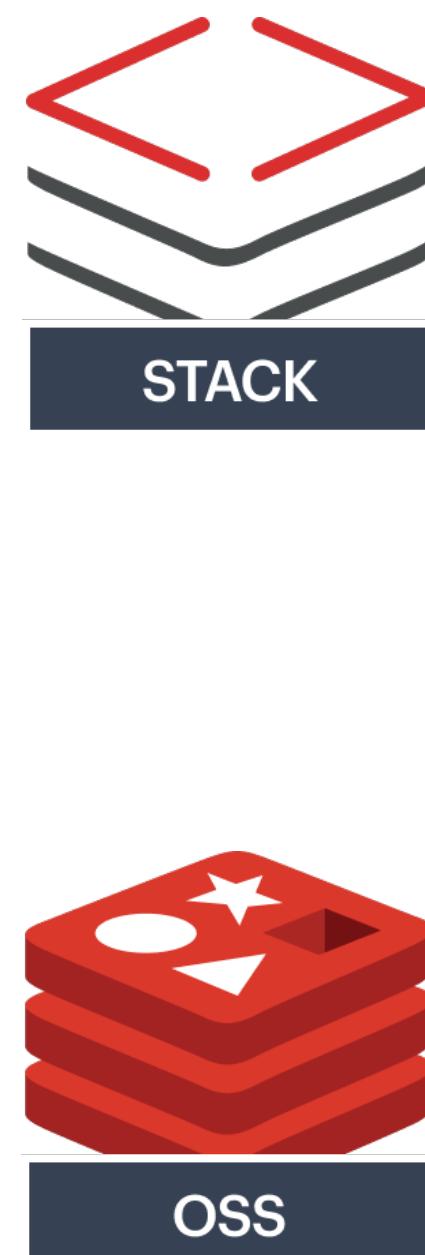
Foundational Data Structures



STACK



OSS



4



3



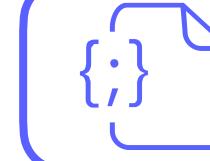
Search



Functions

Redis-side Computing

2



JSON



Probabilistic

Documents and Probabilistic Data Structures

1

Sets

Hashes

Sorted Sets

Bit Field

HyperLogLog

Strings

Lists

Bitmaps

Geospatial

Streams

Foundational Data Structures



STACK

4



redisōM

3



Search



Functions

Redis-side Computing

2



JSON



Probabilistic

Documents and Probabilistic Data Structures

1

Sets

Hashes

Sorted Sets

Bit Field

HyperLogLog

Strings

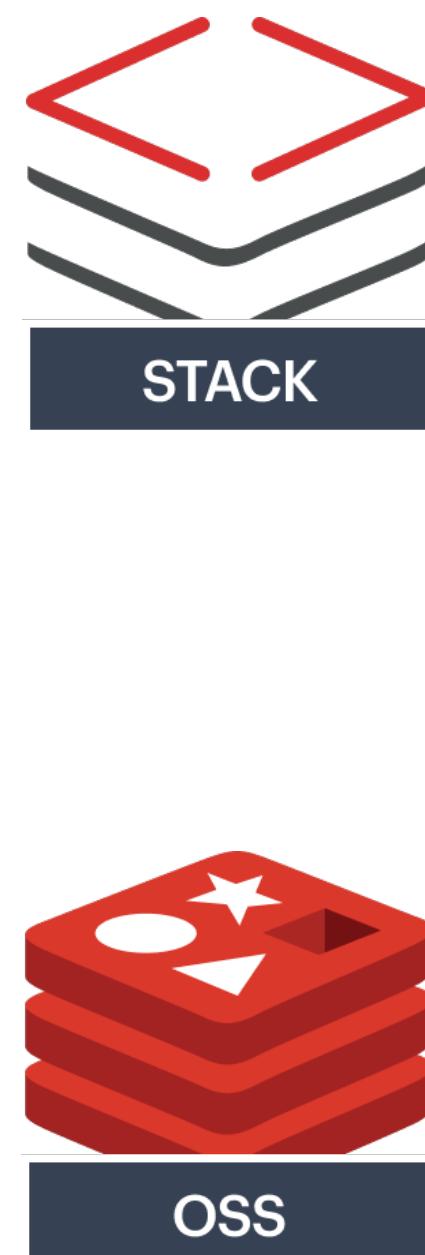
Lists

Bitmaps

Geospatial

Streams

Foundational Data Structures



STACK

4



3



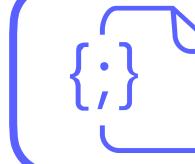
Search



Functions

Redis-side Computing

2



JSON



Probabilistic

Documents and Probabilistic Data Structures

1

Sets

Hashes

Sorted Sets

Bit Field

HyperLogLog

Strings

Lists

Bitmaps

Geospatial

Streams

Foundational Data Structures



STACK

1

Sets

Strings

Hashes

Lists

Sorted Sets

Bitmaps

Bit Field

Geospatial

HyperLogLog

Streams

Foundational Data Structures

2

JSON

Probabilistic

Documents and Probabilistic Data Structures

3

Search

Functions

Redis-side Computing

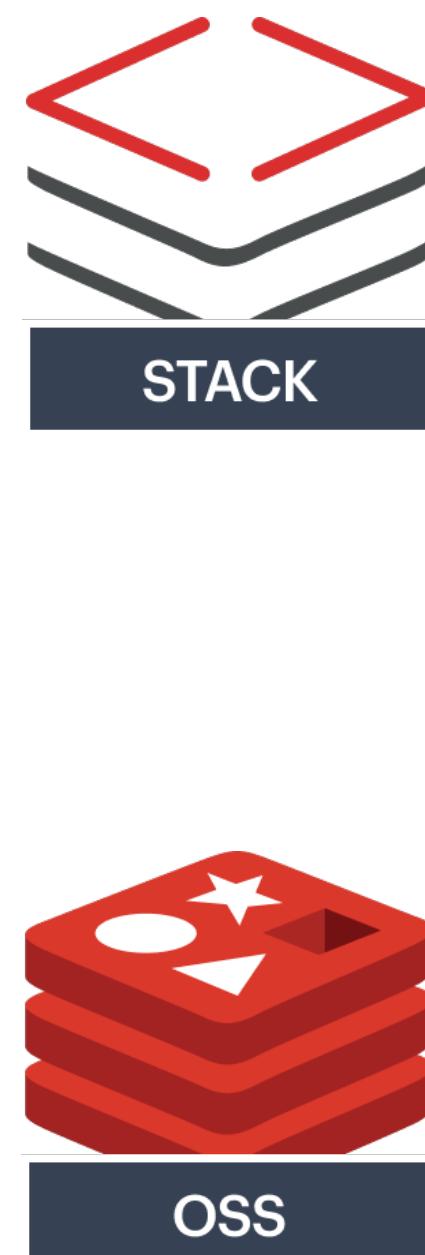
4

redisinsight

redisōM

Jedis

node-redis



STACK

4



3



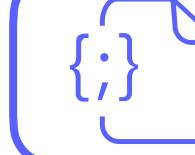
Search



Functions

Redis-side Computing

2



JSON



Probabilistic

Documents and Probabilistic Data Structures

1

Sets

Hashes

Sorted Sets

Bit Field

HyperLogLog

Strings

Lists

Bitmaps

Geospatial

Streams

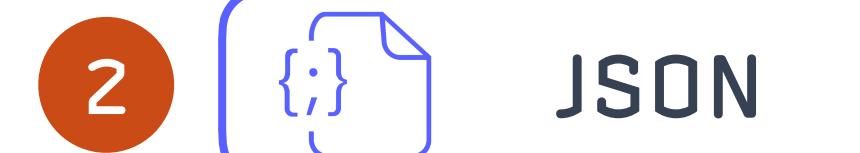
Foundational Data Structures



Developer Experience



Redis-side Computing

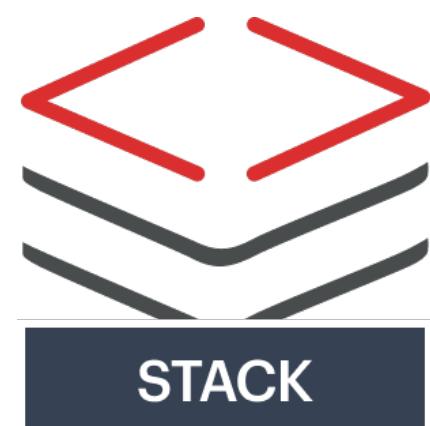


Documents and Probabilistic Data Structures



Foundational Data Structures

Real-Time Data Platform



STACK



OSS



Developer Experience



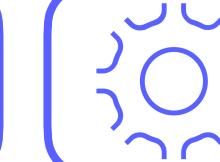
Jedis

node-redis

redis-py



Search



Functions

Redis-side Computing



JSON



Probabilistic

Documents and Probabilistic Data Structures



Sets

Hashes

Sorted Sets

Bit Field

HyperLogLog

Strings

Lists

Bitmaps

Geospatial

Streams

Foundational Data Structures

A Quick Tour of Redis



The Data Balance

Structured vs. Unstructured

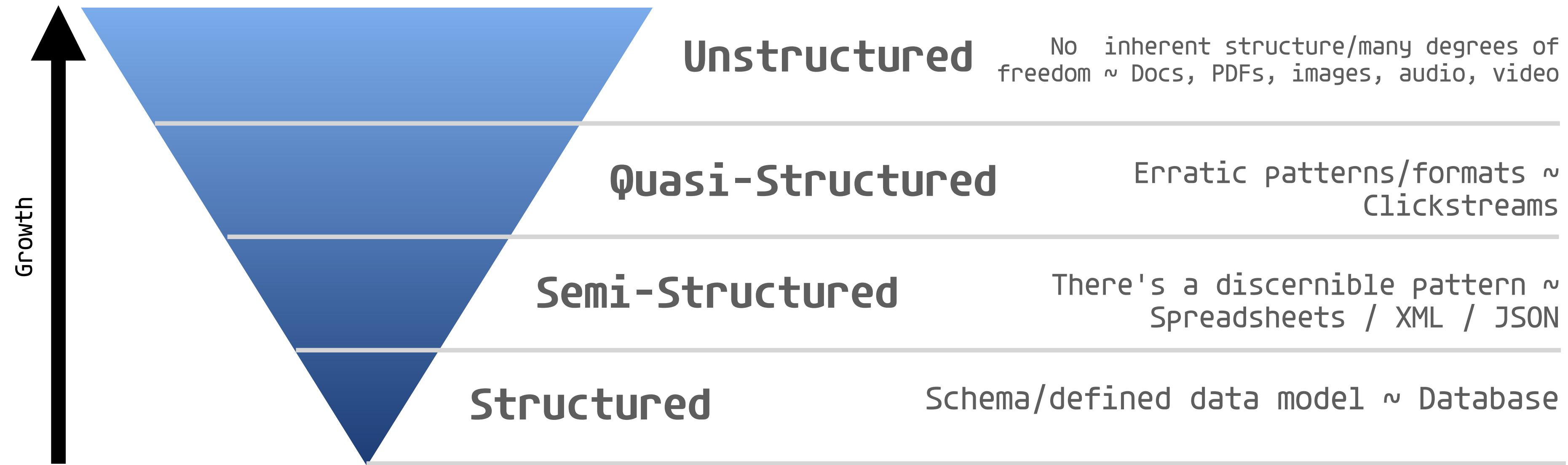
The balanced of data has **changed** radically...

~80% of the data generated by organizations is Unstructured

IDC report, 2020

... and this percentage is estimated to keep growing

with a compound annual growth rate (CAGR) of 36.5% between 2020 and 2025



How to deal with **unstructured** data?

Common approaches were **labeling** and **tagging**

These are **labor intensive, subjective, and error-prone**

Embeddings

Machine Learning Embeddings

Machine Learning/Deep Learning has leaped forward in last decade

ML models **outperform** humans in many tasks nowadays

🔥 CV (Computer Vision) **models** excel at detection/classification

🔥 LLMs (Large Language Models) have advanced exponentially

Machine Learning/Deep Learning has leaped forward in last decade

ML models **outperform** humans in many tasks nowadays

🔥 CV (Computer Vision) **models** excel at detection/classification



LLMs (Large Language Models) have advanced exponentially

Machine Learning/Deep Learning has leaped forward in last decade

ML models **outperform** humans in many tasks nowadays

🔥 CV (Computer Vision) **models** excel at detection/classification



LLMs (Large Language Models) have advanced exponentially

Shut up, Josh!

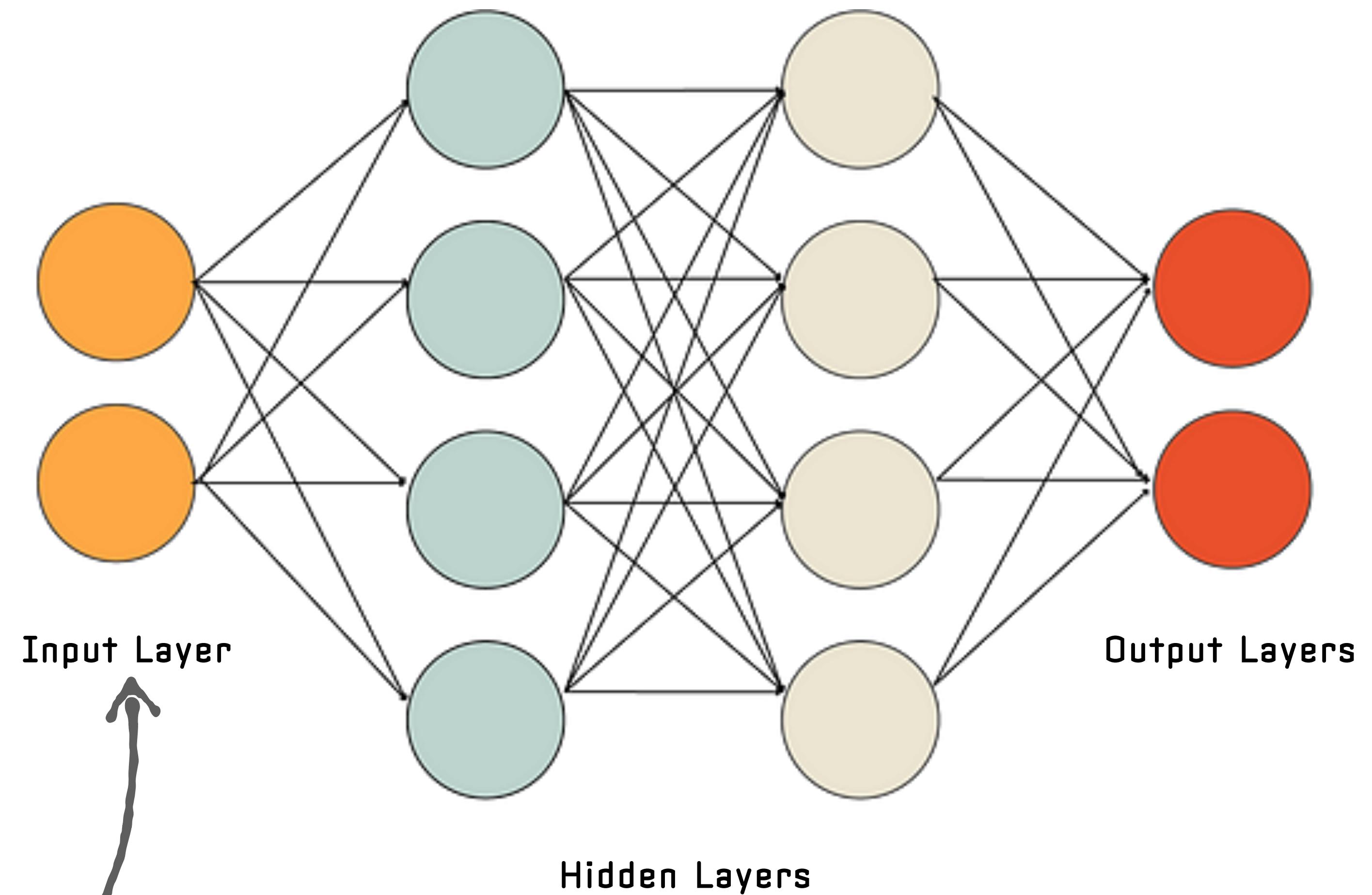
Feature Engineering

Raw Data

Feature	Value
Price	150
Category	Shirt

Scaled and 1-Hot Encoded

Feature	Value
Price	0.45
Category Shirt	1
Category Pants	0
Category Coats	0
Category Shoes	0





Shirt

Jacket

Automated Feature Engineering

ML models extract **latent** features

ML models embeddings catch the **gray** areas between features

The process of generating the embeddings is **vectorizing**

Vectorizing

Generating Vector Embedding for your Data

Vectorizing

Steps to Vectorizing

- 1 Choose an **Embedding Method**
- 2 **Clean** and **preprocess** the **data** as needed
- 3 **Train** the embedding model
- 4 Generate **Embeddings**

Vectorizing

Better Models, better Vectors

Embeddings can capture the **semantics** of complex data

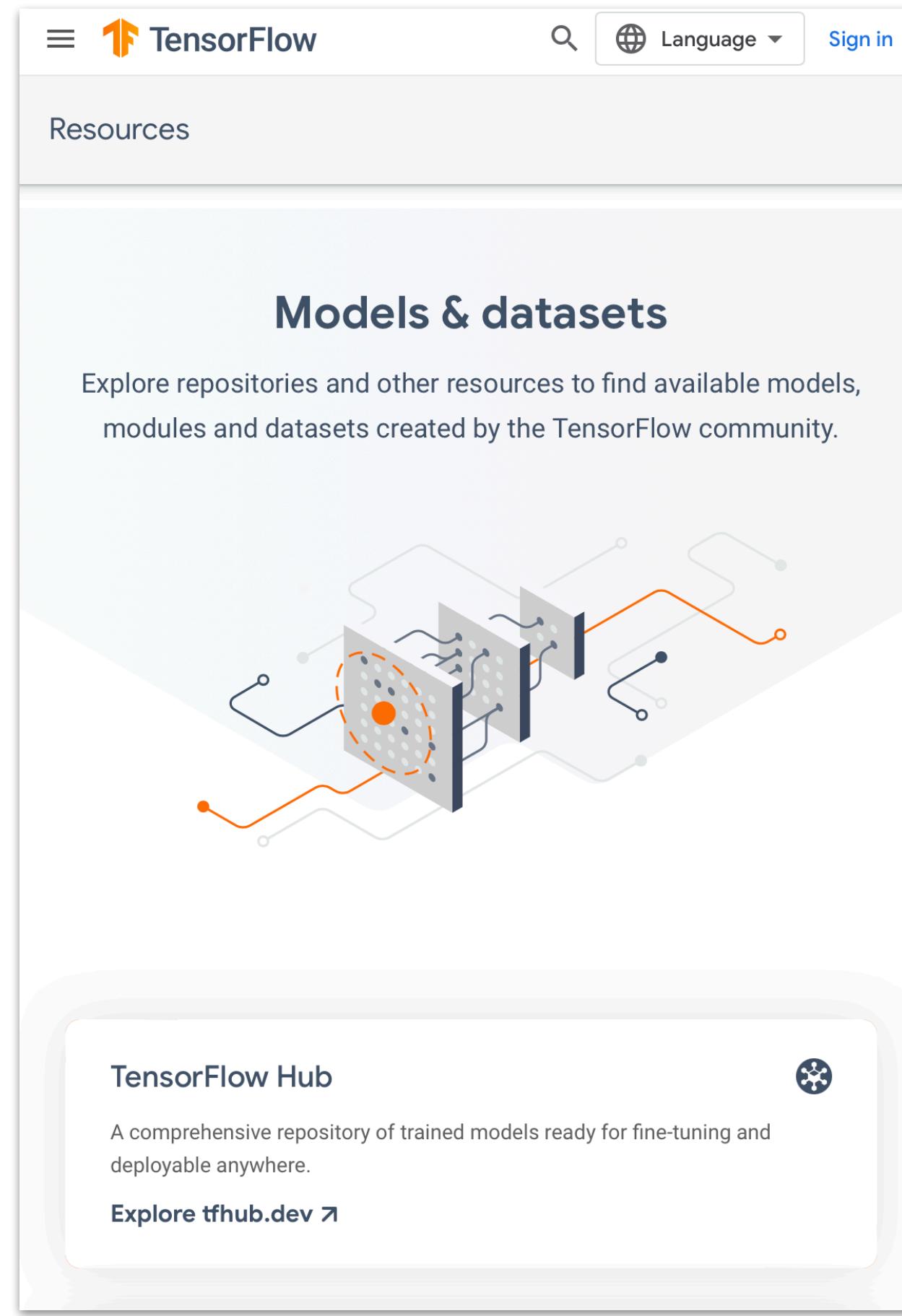
Option #1: Use a **pre-trained** model

Option #2: **train** your models with **custom** data

Vector similarity is a downline tool to **analyze embeddings**

Vectorizing

Ever growing collection of pre-trained, trainable and scriptable models



TensorFlow Language Sign in

Resources

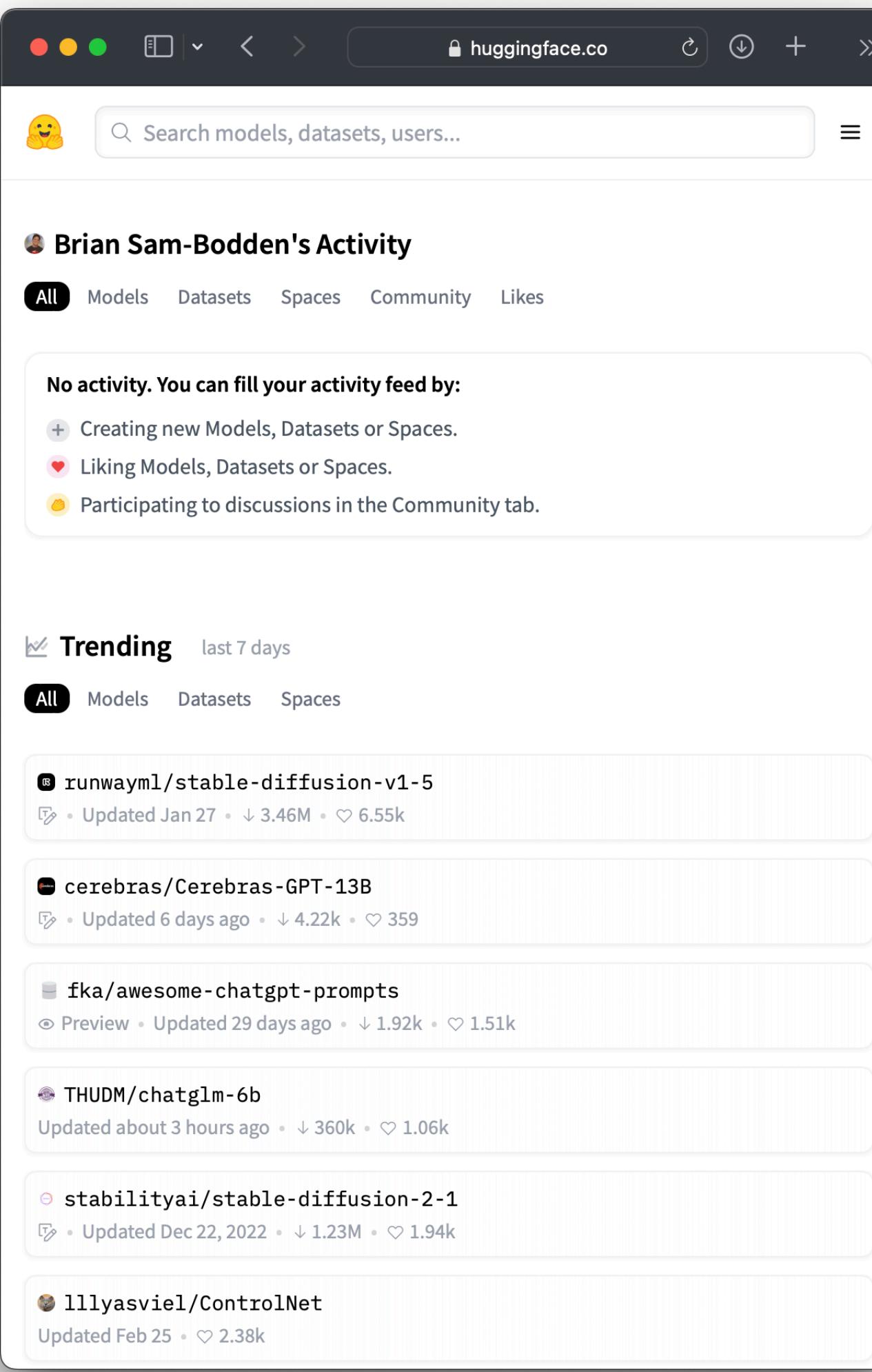
Models & datasets

Explore repositories and other resources to find available models, modules and datasets created by the TensorFlow community.

A comprehensive repository of trained models ready for fine-tuning and deployable anywhere.

TensorFlow Hub 🔗

Explore [tflib.dev](#) ↗



Search models, datasets, users...

Brian Sam-Bodden's Activity

All Models Datasets Spaces Community Likes

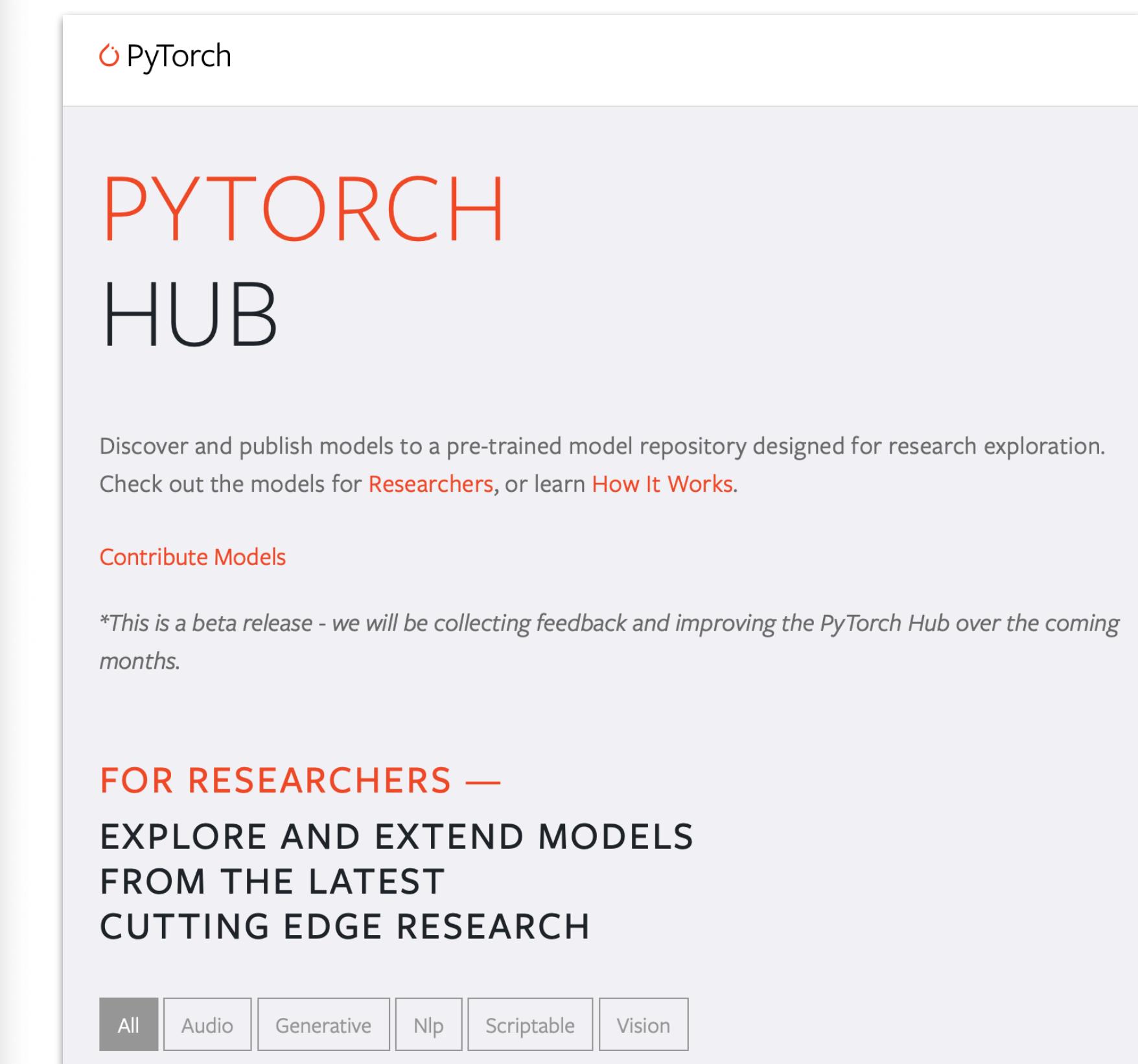
No activity. You can fill your activity feed by:

- + Creating new Models, Datasets or Spaces.
- Heart Liking Models, Datasets or Spaces.
- 💡 Participating to discussions in the Community tab.

Trending last 7 days

All Models Datasets Spaces

- runwayml/stable-diffusion-v1-5 · Updated Jan 27 · 3.46M · 6.55k
- cerebras/Cerebras-GPT-13B · Updated 6 days ago · 4.22k · 359
- fka/awesome-chatgpt-prompts · Preview · Updated 29 days ago · 1.92k · 1.51k
- THUDM/chatglm-6b · Updated about 3 hours ago · 360k · 1.06k
- stabilityai/stable-diffusion-2-1 · Updated Dec 22, 2022 · 1.23M · 1.94k
- llyasviel/ControlNet · Updated Feb 25 · 2.38k



PyTorch

PYTORCH HUB

Discover and publish models to a pre-trained model repository designed for research exploration. Check out the models for [Researchers](#), or learn [How It Works](#).

Contribute Models

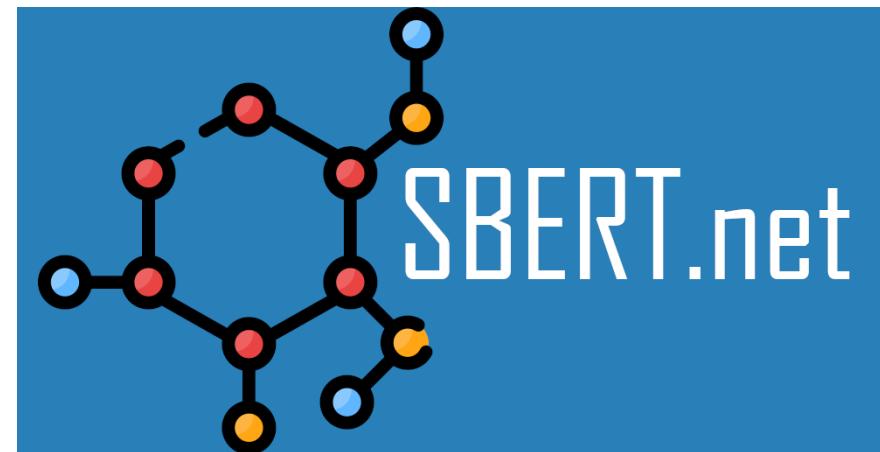
*This is a beta release - we will be collecting feedback and improving the PyTorch Hub over the coming months.

FOR RESEARCHERS — EXPLORE AND EXTEND MODELS FROM THE LATEST CUTTING EDGE RESEARCH

All Audio Generative Nlp Scriptable Vision

Vectorizing

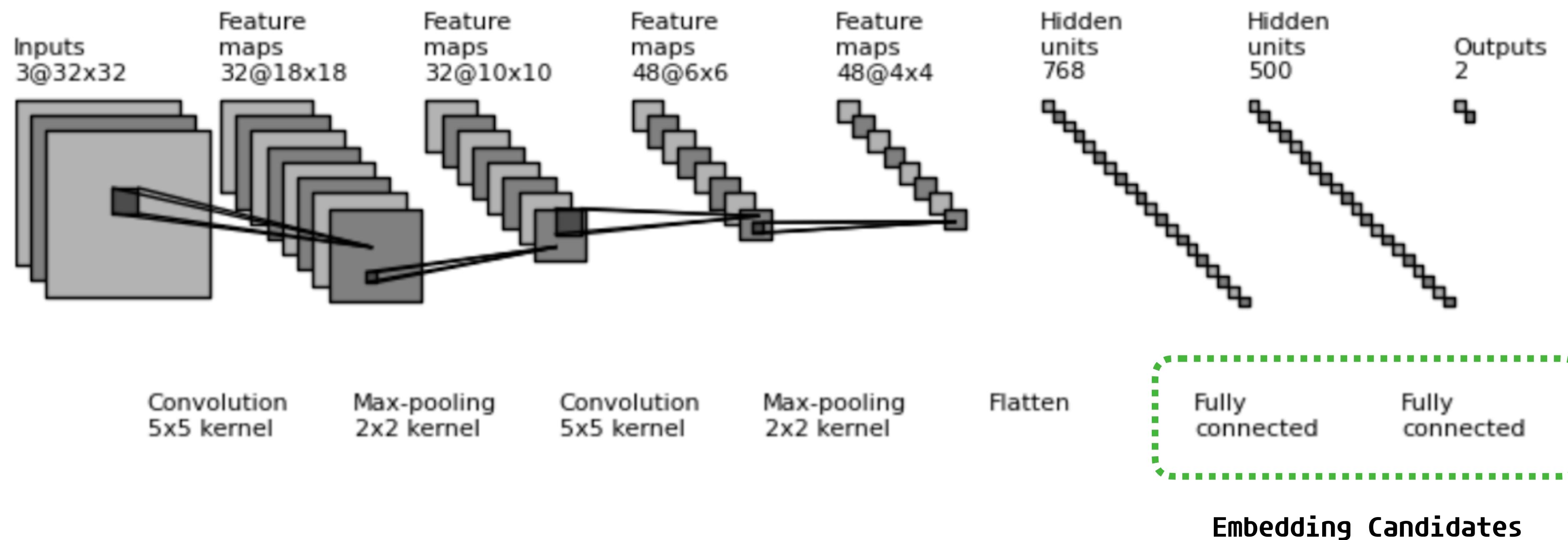
For sentences, [SBERT.net](#) provides a variety of pre-trained models:



Model Name	Performance Sentence Embeddings (14 Datasets) <small>i</small>	Performance Semantic Search (6 Datasets) <small>i</small>	Avg.		
			Performance	Speed	Model Size <small>i</small>
all-mpnet-base-v2 <small>i</small>	69.57	57.02	63.30	2800	420 MB
multi-qa-mpnet-base-dot-v1 <small>i</small>	66.76	57.60	62.18	2800	420 MB
all-distilroberta-v1 <small>i</small>	68.73	50.94	59.84	4000	290 MB
all-MiniLM-L12-v2 <small>i</small>	68.70	50.82	59.76	7500	120 MB
multi-qa-distilbert-cos-v1 <small>i</small>	65.98	52.83	59.41	4000	250 MB
all-MiniLM-L6-v2 <small>i</small>	68.06	49.54	58.80	14200	80 MB
multi-qa-MiniLM-L6-cos-v1 <small>i</small>	64.33	51.83	58.08	14200	80 MB
paraphrase-multilingual-mpnet-base-v2 <small>i</small>	65.83	41.68	53.75	2500	970 MB
paraphrase-albert-small-v2 <small>i</small>	64.46	40.04	52.25	5000	43 MB
paraphrase-multilingual-MiniLM-L12-v2 <small>i</small>	64.25	39.19	51.72	7500	420 MB
paraphrase-MiniLM-L3-v2 <small>i</small>	62.29	39.19	50.74	19000	61 MB
distiluse-base-multilingual-cased-v1 <small>i</small>	61.30	29.87	45.59	4000	480 MB
distiluse-base-multilingual-cased-v2 <small>i</small>	60.18	27.35	43.77	4000	480 MB

Vectorizing

Extracted a 1-dimensional layer that's **densely packed** with information about present features



Vectors

Storing and creating Vectors

Vectors

What's is a Vector?

Numeric representation of something in N-dimensional space

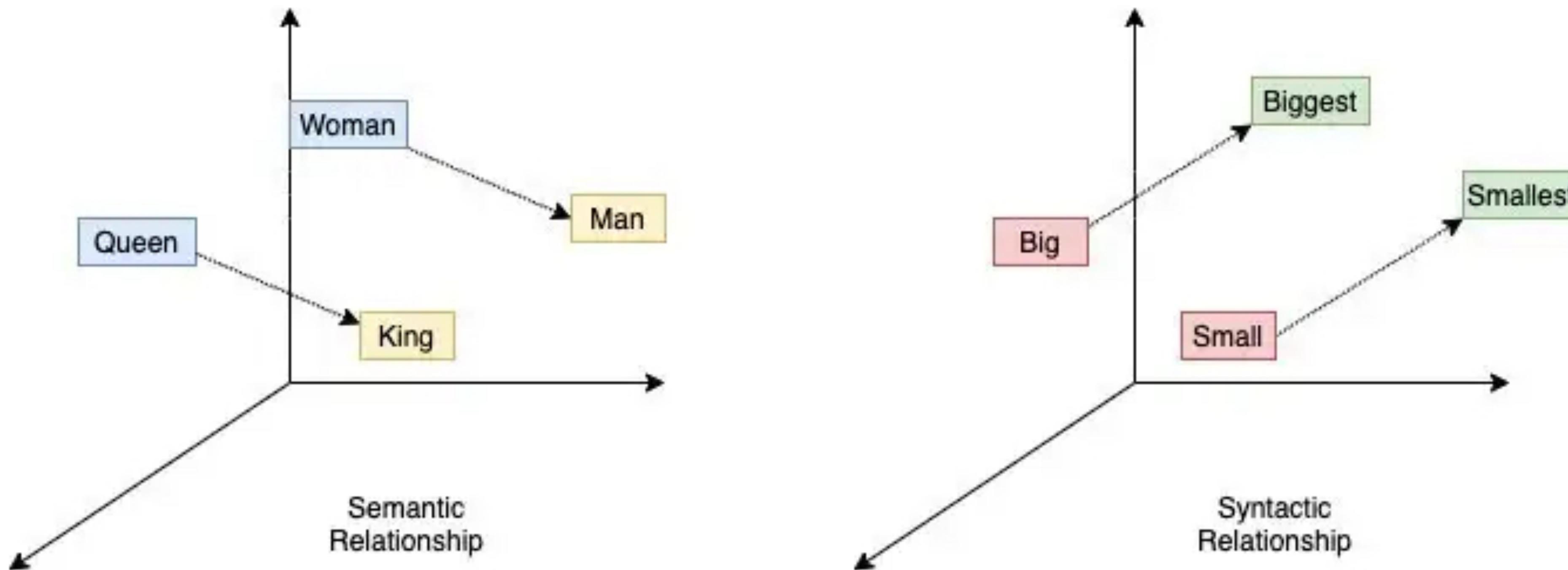
Can represent anything... entire documents, images, video, audio

Quantifies features or characteristics of the item

More importantly... they are comparable

Vectors

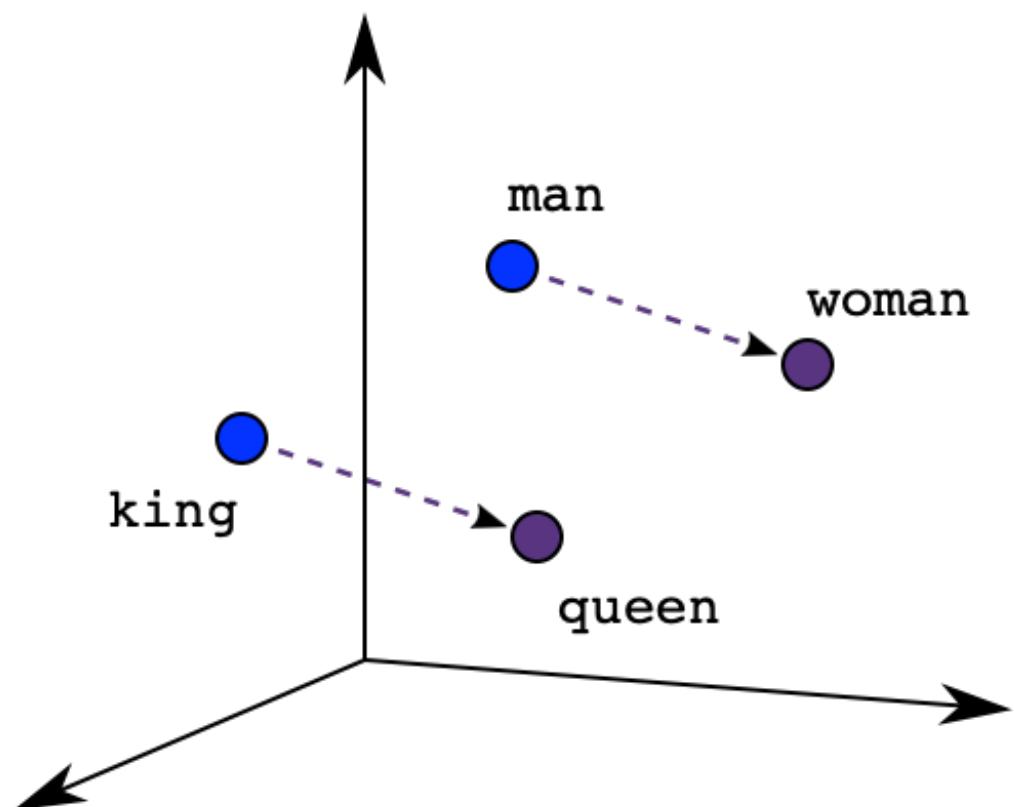
Visually



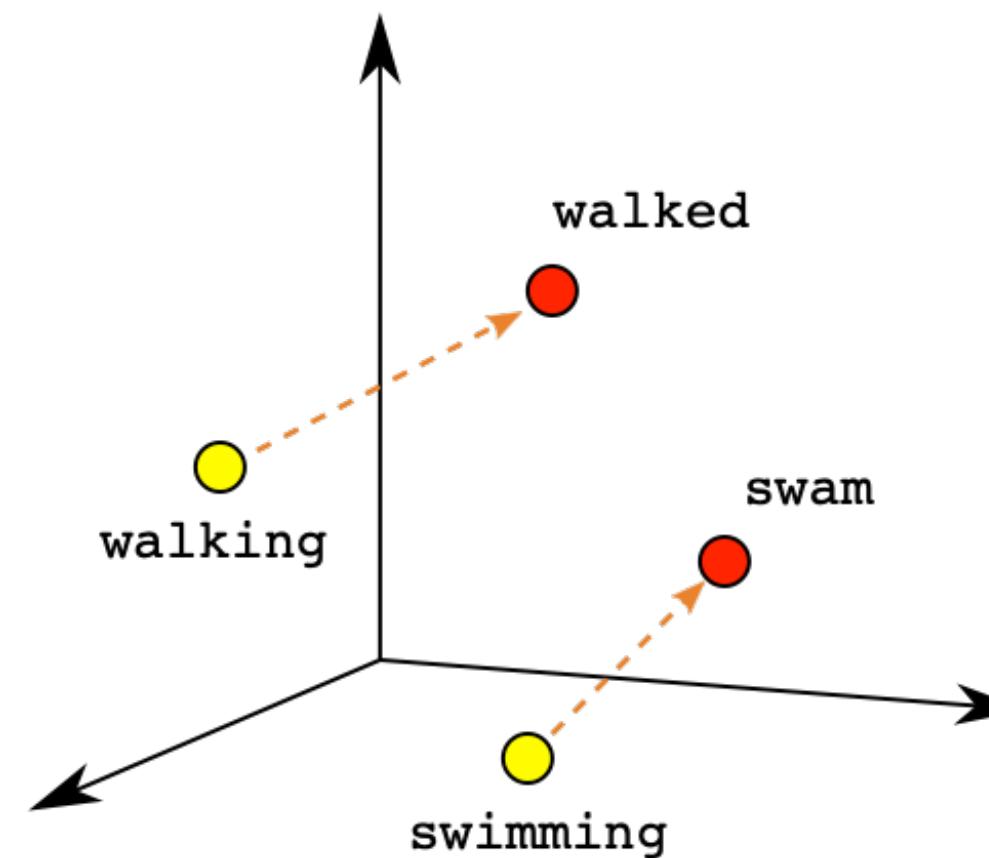
Vectors

Visually

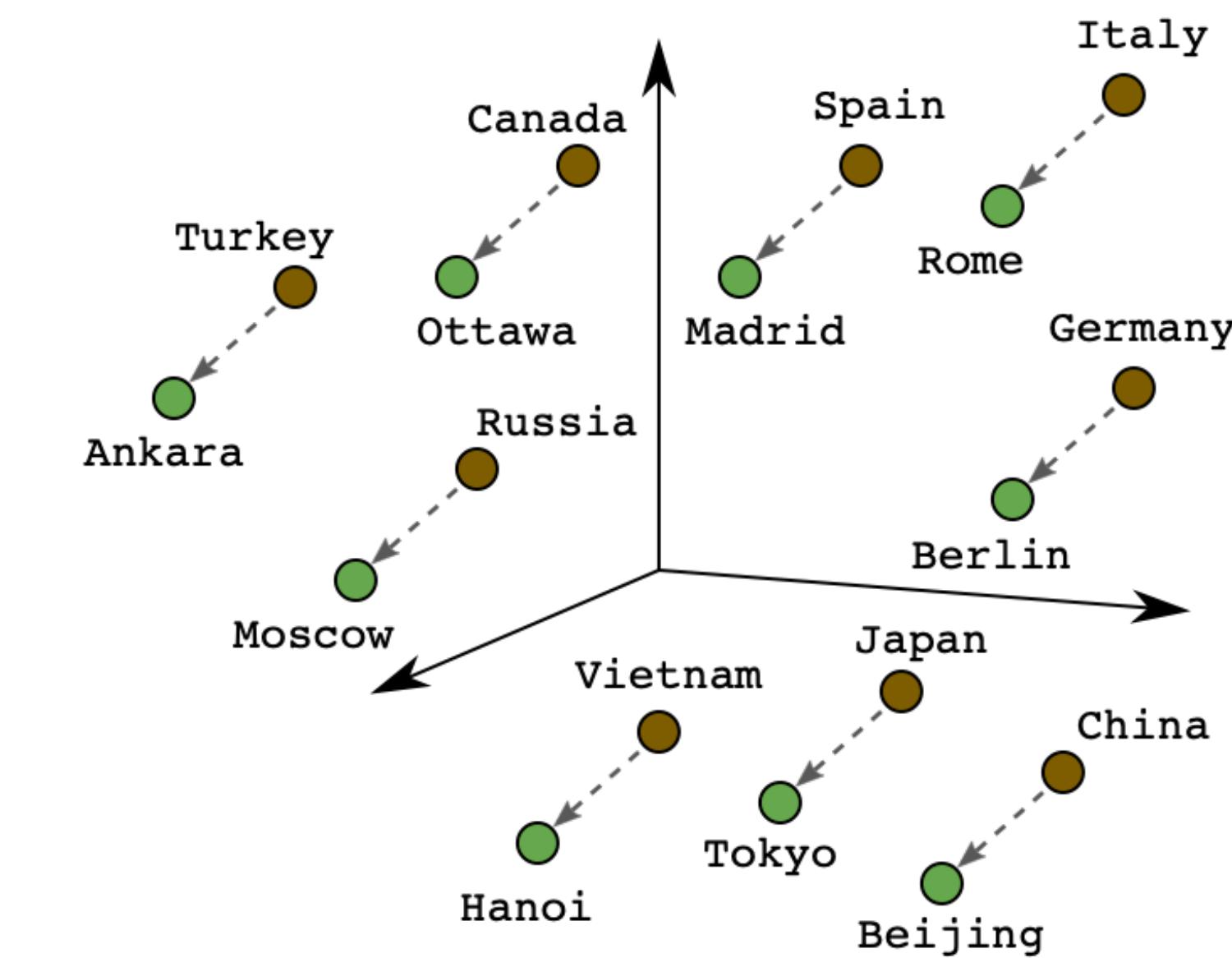
Male-Female



Verb Tense



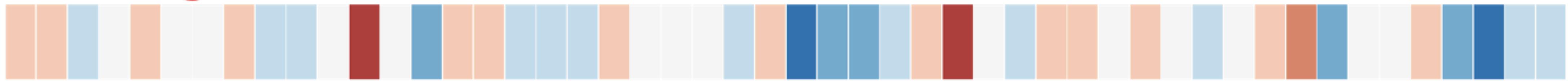
Country-Capital



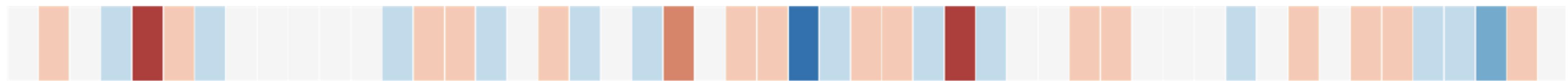
Vectors

Visually

“king”



“Man”



“Woman”



Let's make some Vectors...





Machine Learning in Java

with DJL - <https://djl.ai>

A **Java** Framework for Machine Learning

Build, train, deploy ML and DL models

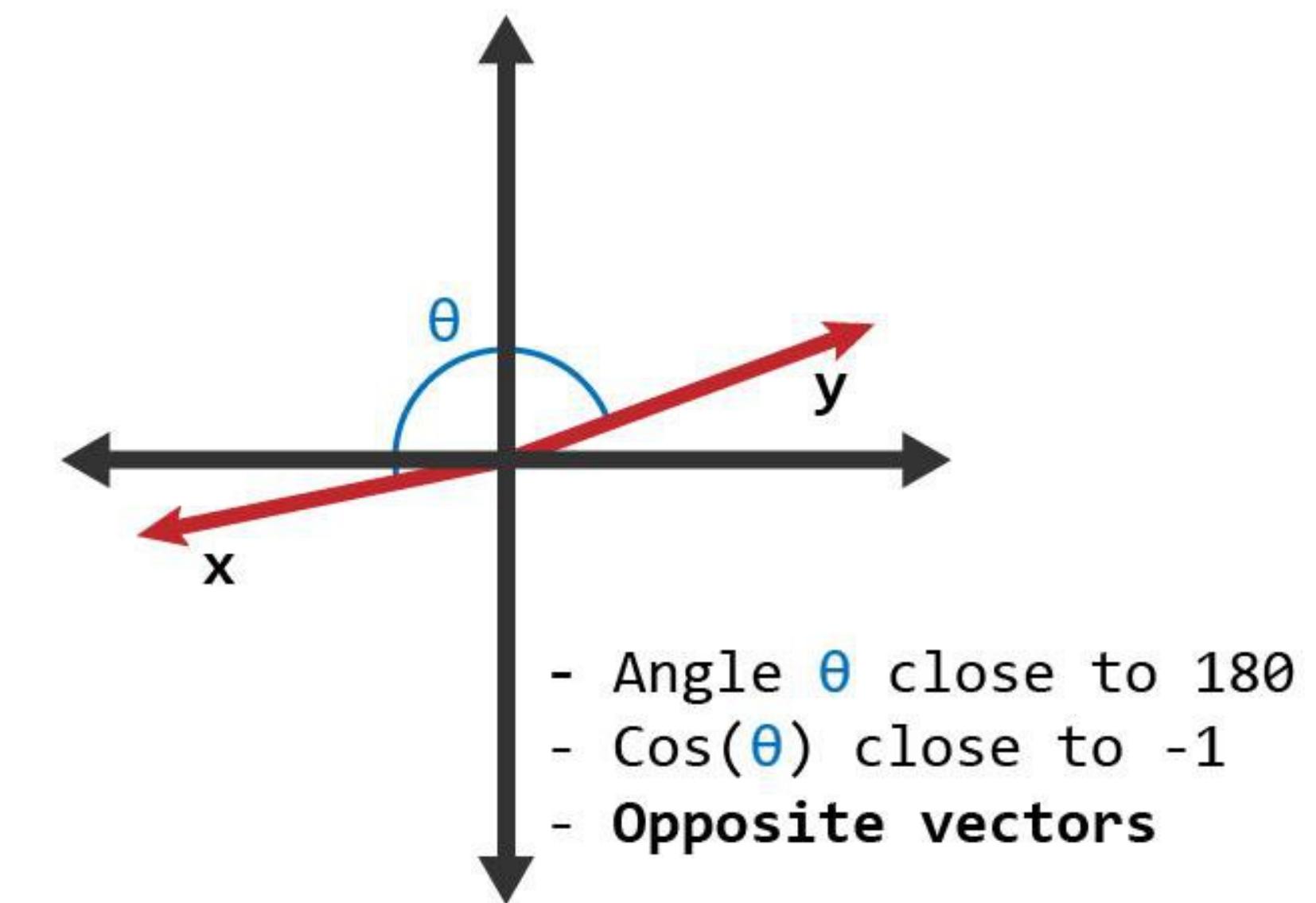
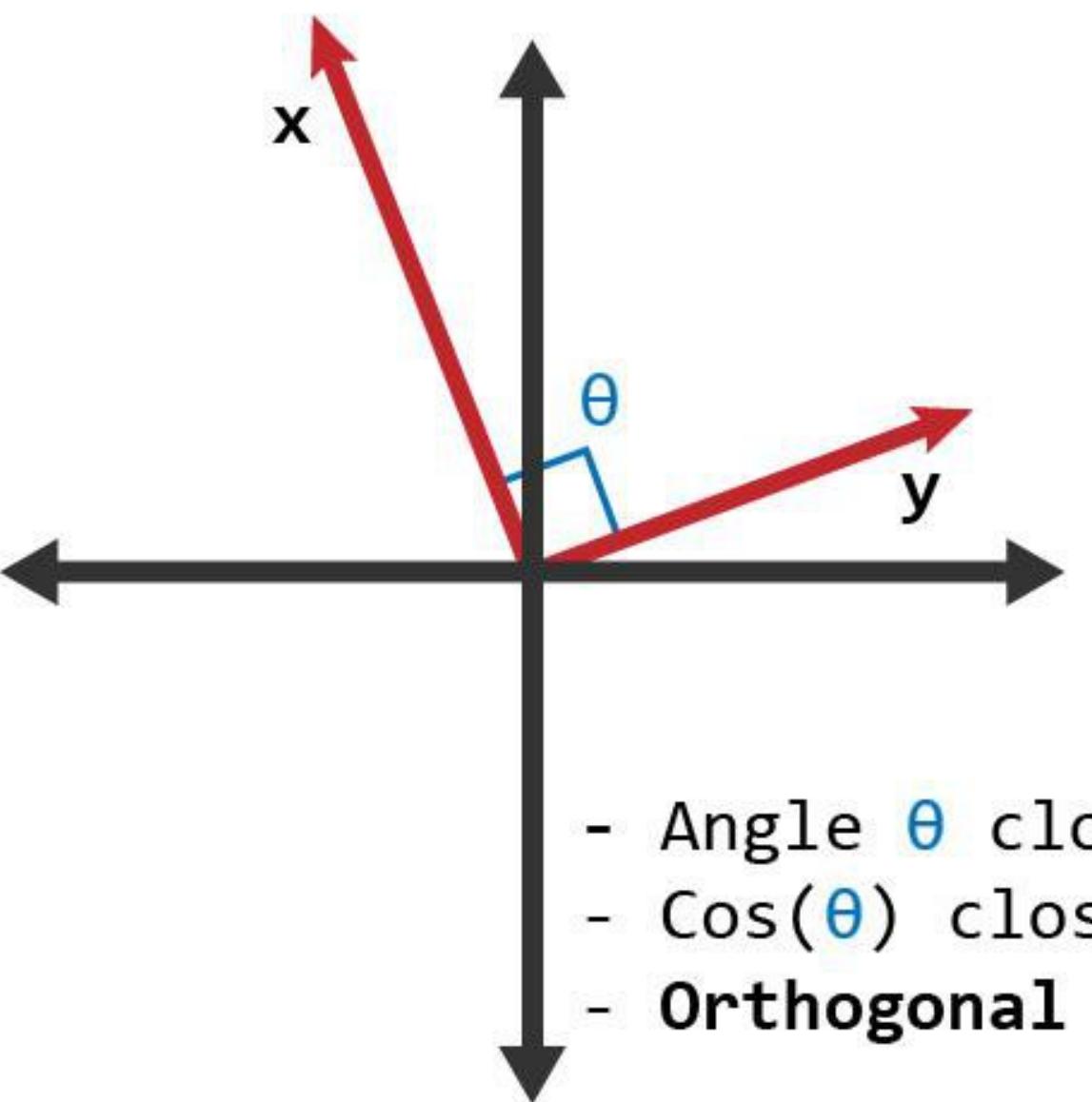
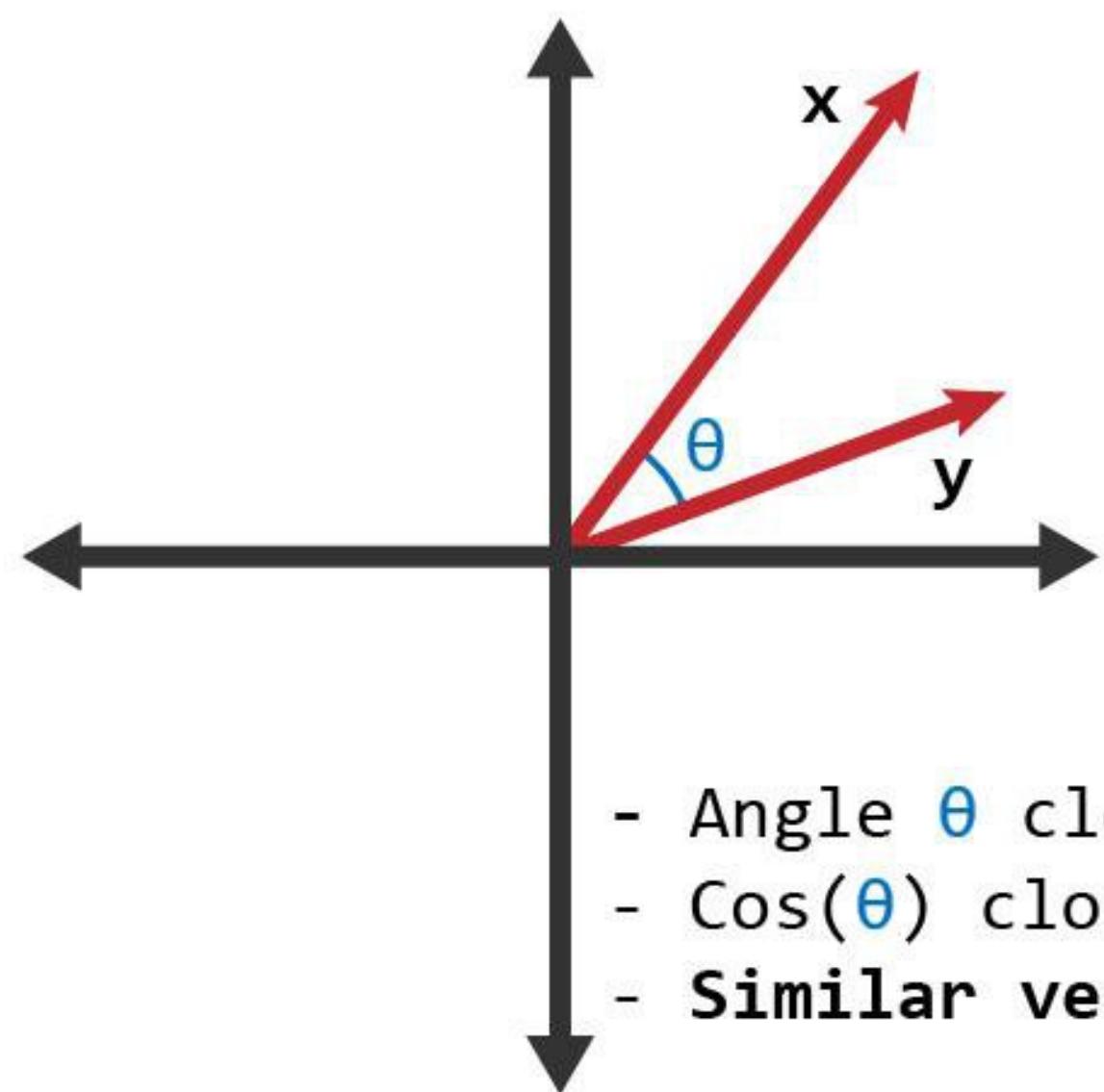
Abstracts **PyTorch**, **Apache MXNet**, **TensorFlow** and **ONNX**

Opens the world of "**Model Zoos**" to the Java Community

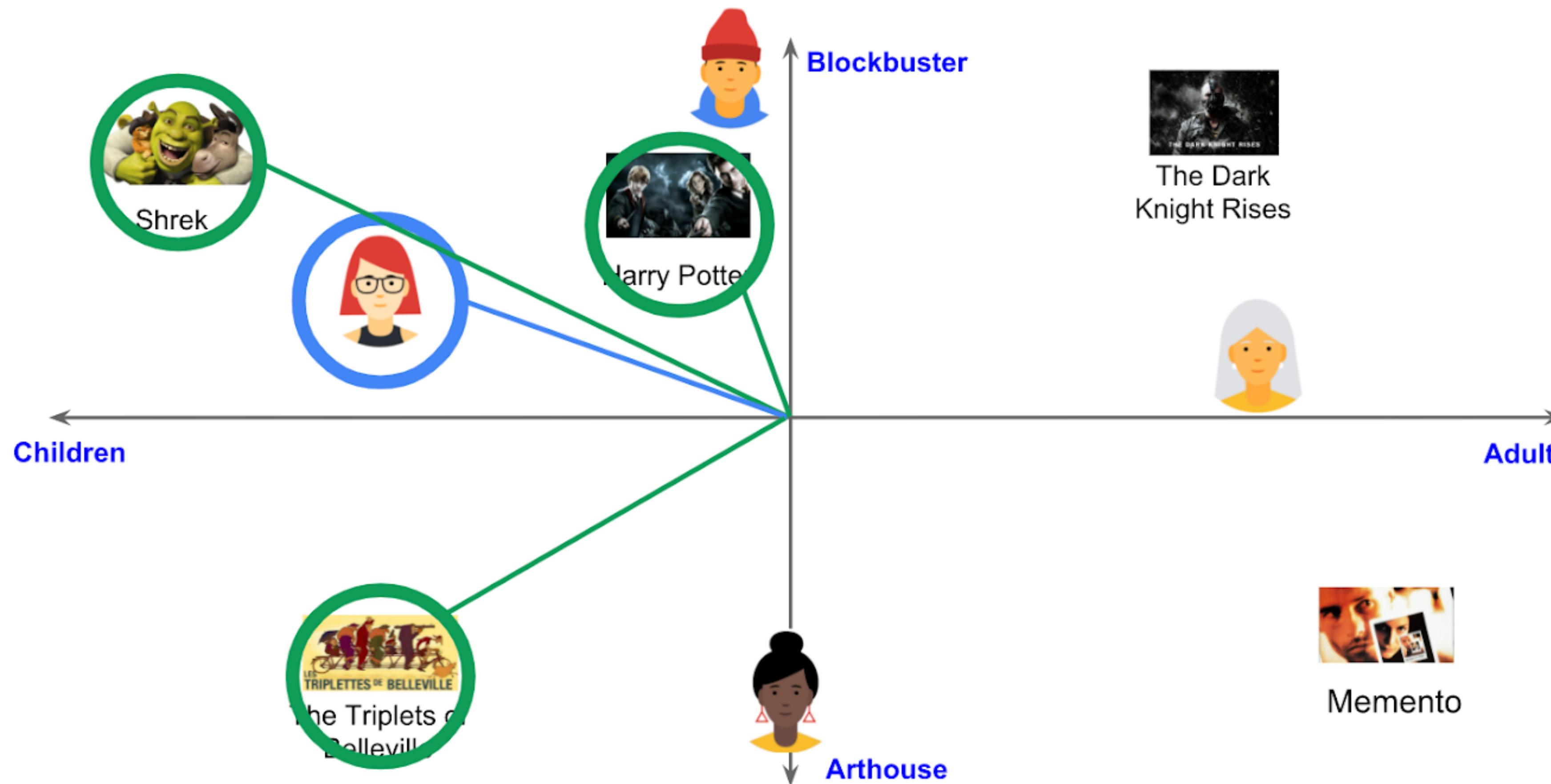
Similarity

Finding similar vectors

Measuring Similarity with Cosines

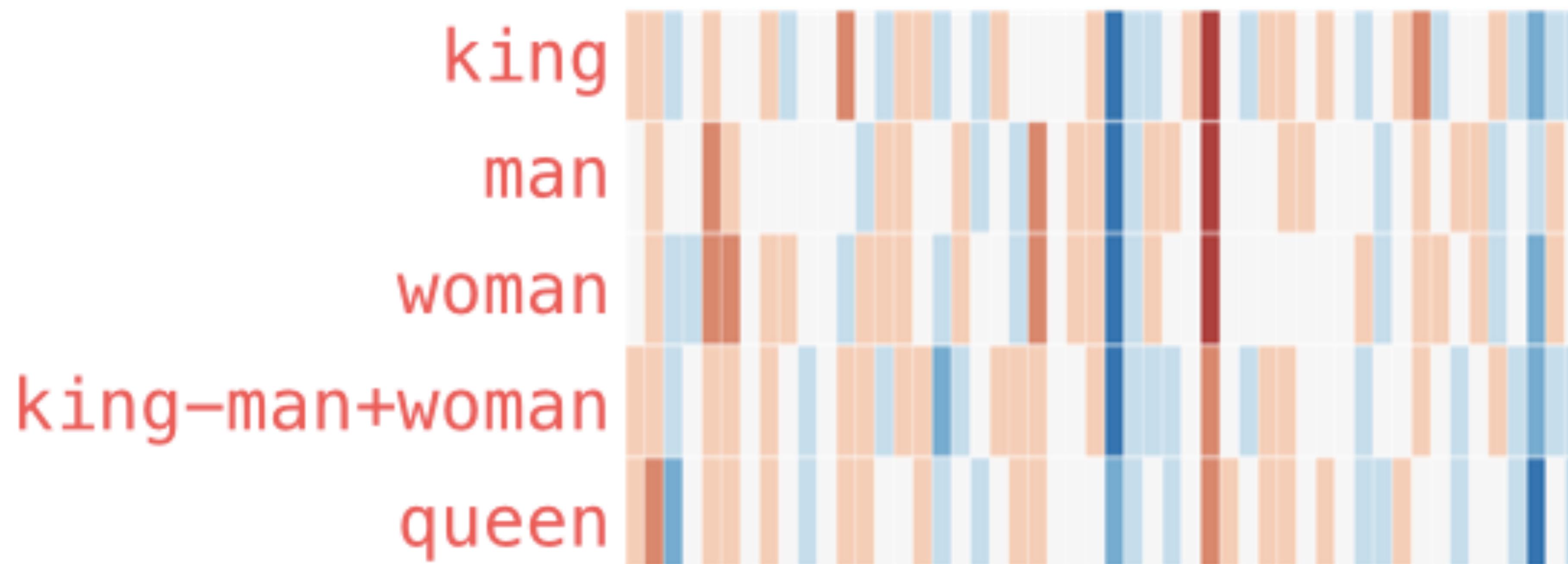


An example of a 2D embedding space for Movies

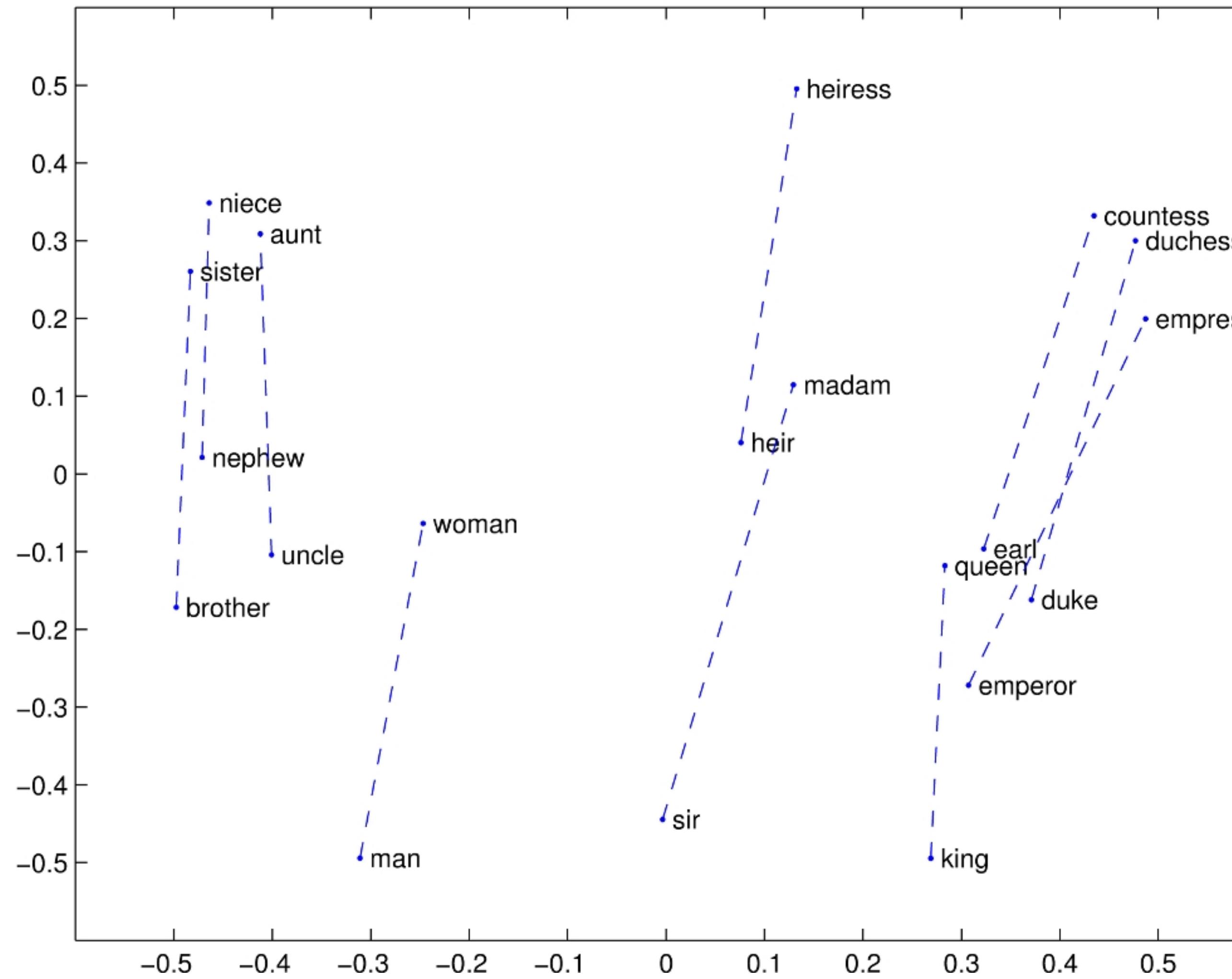


Vectors can be operated upon

$$\text{king} - \text{man} + \text{woman} \approx \text{queen}$$

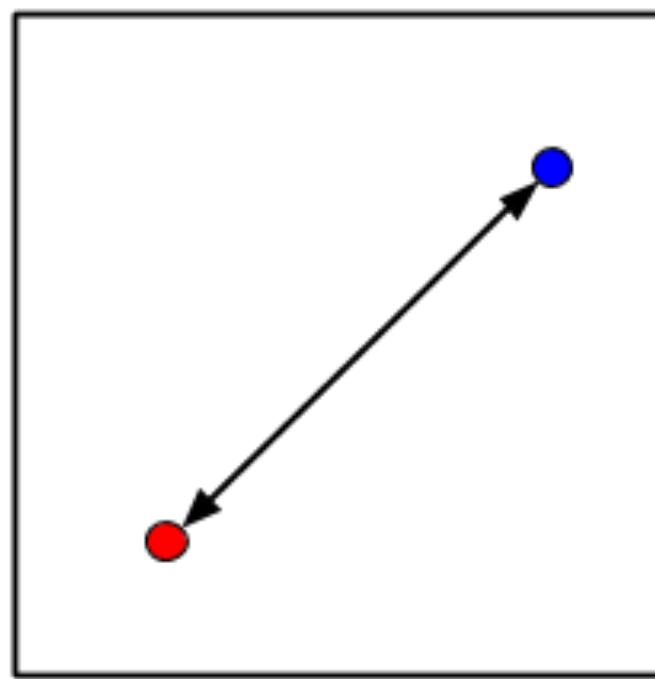


Vector Difference between Pairs of Word Vectors

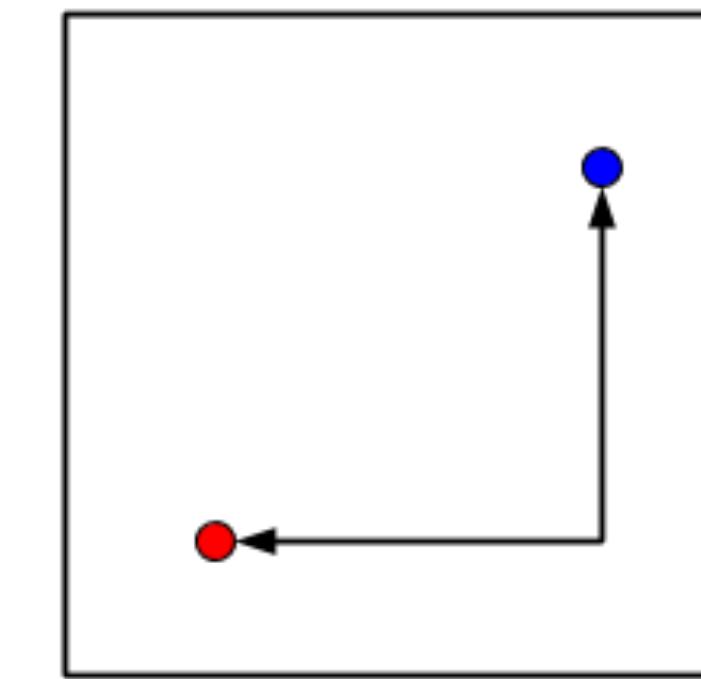


A few of the many Distance Metrics

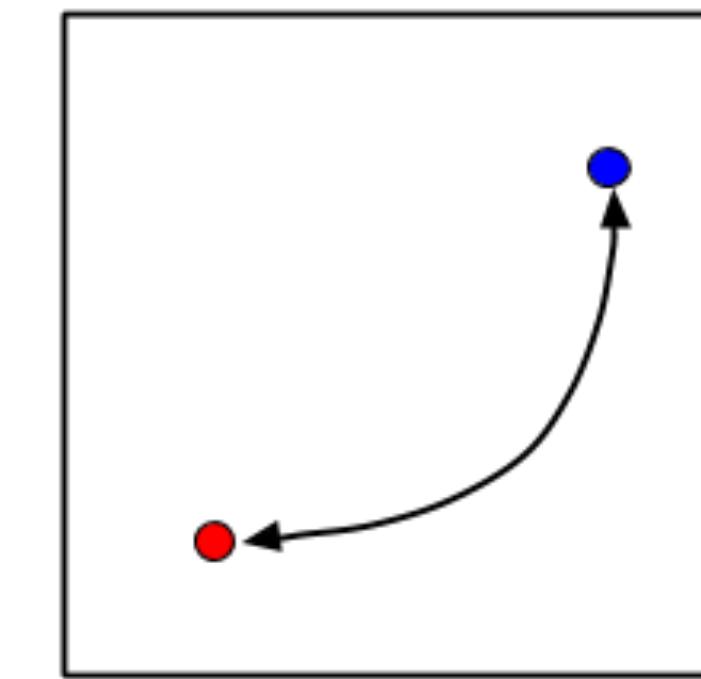
Euclidean



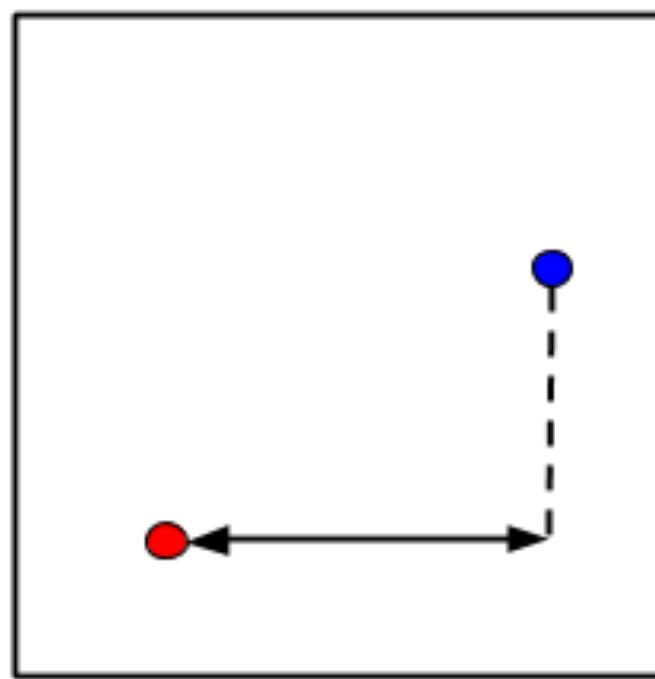
Manhattan



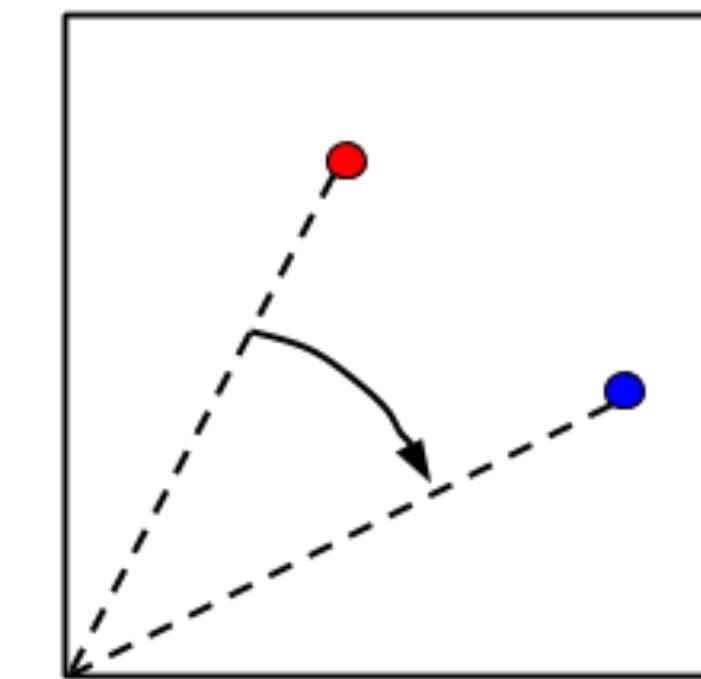
Minkowski



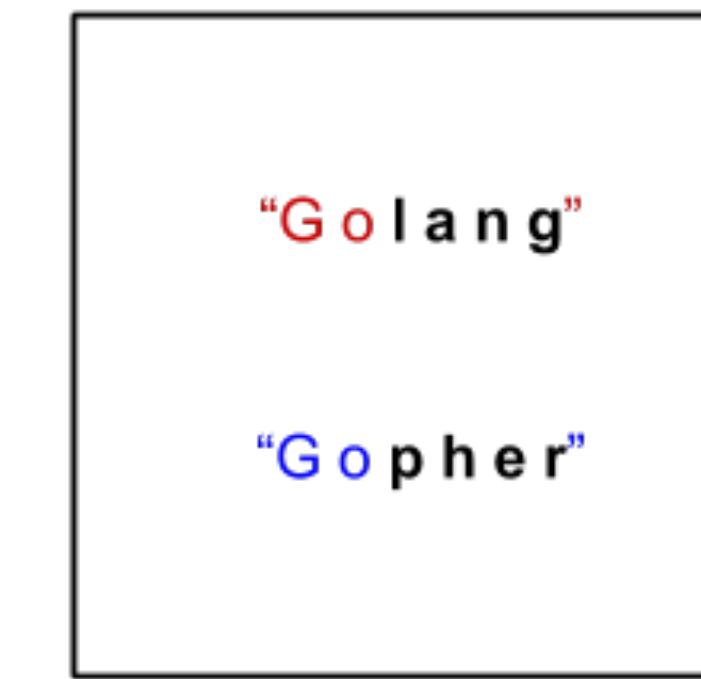
Chebychev



Cosine Similarity



Hamming



Redis OM Spring

Extending Spring Data Redis

Multi-Model Object-Mapping & Querying



Multi-Model Object-Mapping & Querying



Spring Data Redis

Multi-Model Object-Mapping & Querying



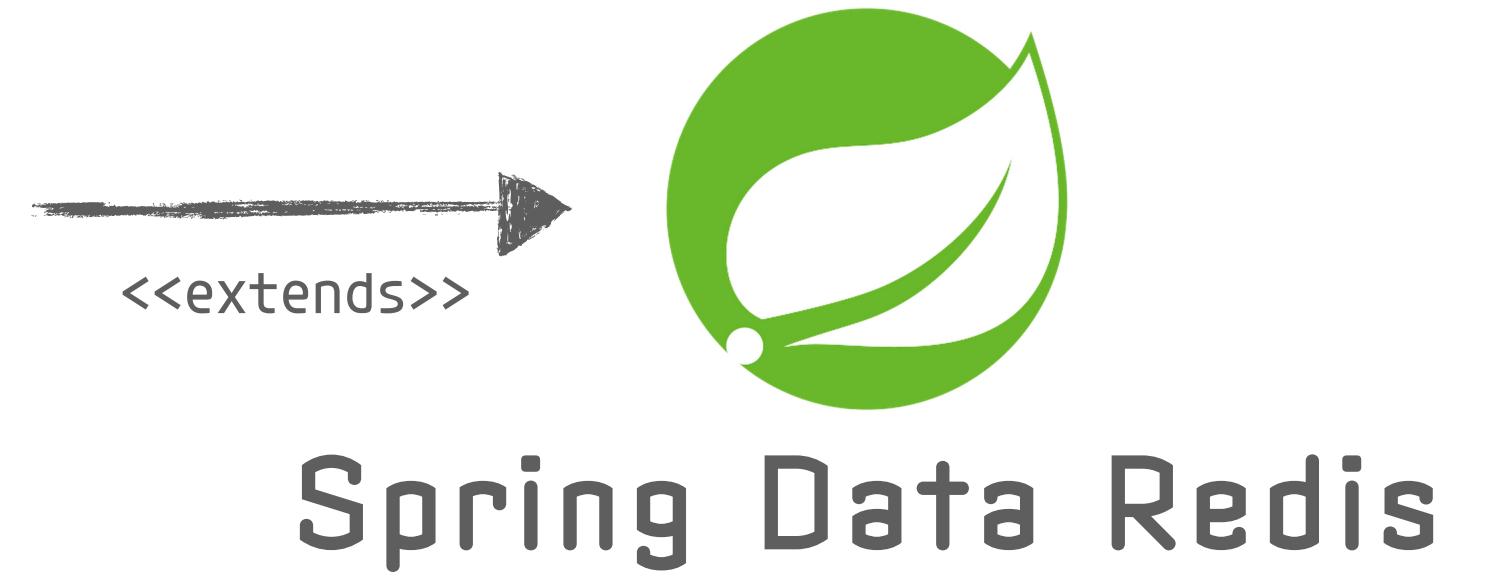
Spring Data Redis

Multi-Model Object-Mapping & Querying



Jedis

redisTM
Redis OM Spring



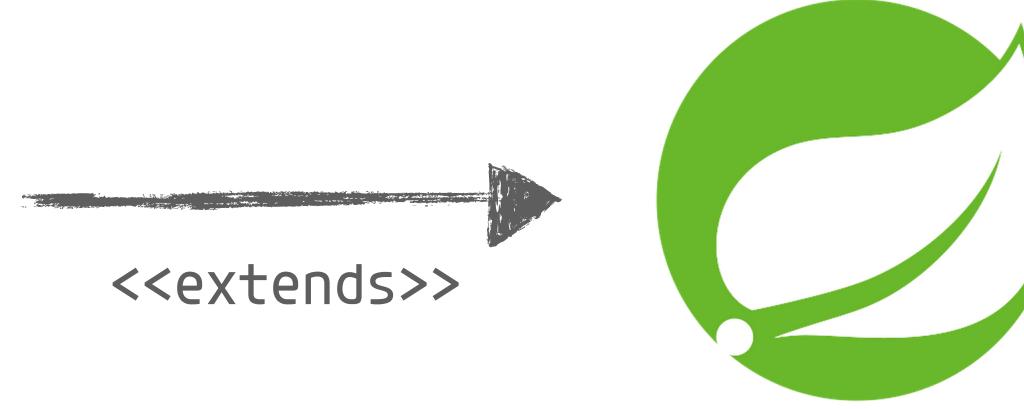
Multi-Model Object-Mapping & Querying



Jedis



redis™
Redis OM Spring

The Redis logo, featuring the word "redis" in red and "TM" in dark blue, with a green leaf icon integrated into the letter "i". Below it, the text "Redis OM Spring" is written in a smaller, gray sans-serif font.

Spring Data Redis

Model → Redis Hash

@RedisHash Annotation

```
@Data  
@Builder  
@RedisHash  
public class Role {  
    @Id  
    private String id;  
  
    private String name;  
}
```

HASH com.redis.stack.demo.models.hashes.Role:01GF0T8VD4FRNW87BCS7JN8JA3		X
232 B	Length: 3	TTL: No limit
Field		Value
_class	com.redis.stack.demo.models.hashes.Role	 
id	01GF0T8VD4FRNW87BCS7JN8JA3	 
name	editor	 

Spring Data Repositories

From Spring Data Redis

```
@Repository  
public interface RoleRepository  
    extends CrudRepository<Role, String> {  
}
```

Method and Description
count() Returns the number of entities available.
delete(T entity) Deletes a given entity.
deleteAll() Deletes all entities managed by the repository.
deleteAll(Iterable<? extends T> entities) Deletes the given entities.
deleteAllById(Iterable<? extends ID> ids) Deletes all instances of the type T with the given IDs.
deleteById(ID id) Deletes the entity with the given id.
existsById(ID id) Returns whether an entity with the given id exists.
findAll() Returns all instances of the type.
findAllById(Iterable<ID> ids) Returns all instances of the type T with the given IDs.
findById(ID id) Retrieves an entity by its id.
save(S entity) Saves a given entity.
saveAll(Iterable<S> entities) Saves all given entities.



Search Index Generation

@Indexed Annotation

```
@Data  
@Builder  
@RedisHash  
public class Role {  
    @Id  
    private String id;  
  
    @Indexed  
    private String name;  
}
```

```
@Repository  
public interface RoleRepository  
    extends CrudRepository<Role, String> {  
  
    Optional<Role> findFirstByName(String name);  
}
```

More Complex & Searchable Models

@Searchable / @Indexed / @Bloom Annotation

```
@Data  
@RedisHash  
public class User {  
    @Id private String id;  
  
    @Searchable private String name;  
  
    @Bloom(name = "bf_company_email", capacity = 100000, errorRate = 0.001)  
    @Indexed  
    private String email;  
    private String password;  
  
    @Transient private String passwordConfirm;  
    @Reference private Set<Role> roles;  
  
    // audit fields  
    @CreatedDate private Date createdDate;  
    @LastModifiedDate private Date lastModifiedDate;  
}
```

More Complex & Searchable Models

@Searchable / @Indexed / @Bloom Annotation

```
@Repository
public interface UserRepository extends CrudRepository<User, String> {
    Iterable<User> findByNameStartingWith(String prefix);

    Optional<User> findFirstByEmail(String email);

    boolean existsByEmail(String email);
}
```

Model → Redis JSON

Document Annotation

```
@Document
public class FictionalCharacter {
    @Id
    @Indexed private String id;

    // Indexed for exact text matching
    @Indexed private String actorFirstName;
    @Indexed private String actorLastName;

    // Indexed for numeric matches
    @Indexed private Integer actorAge;

    // Indexed for Full Text matches
    @Searchable private String quote;

    // Indexed for Geo Filtering
    @Indexed private Point actorLocation;

    // Nest indexed object
    @Indexed private Address actorAddress;

    @Indexed private Set<String> skills;
}
```

JSON com.redis.stack.demo.models.json.FictionalCharacter:01GF0T90HQTCTSSD772B26N885  

Key Size: 554 B Length: 8 TTL: No limit  

```
{
    "id": "01GF0T90HQTCTSSD772B26N885",
    "actorFirstName": "Zoe",
    "actorLastName": "Saldana",
    "actorAge": 43,
    "quote": "I Am Going To Die Surrounded By The Biggest Idiots In The Galaxy.",
    "actorLocation": "-118.399968,34.073087",
    "actorAddress": {
        "houseNumber": "107",
        "street": "S Beverly Glen Blvd",
        "city": "Los Angeles",
        "state": "CA",
        "postalCode": "90024",
        "country": "US"
    },
    "skills": [
        "0": "martial_arts",
        "1": "skills"
    ]
}
```

More Repository Superpowers w/ JSON

@Searchable / @Indexed

```
public interface FictionalCharacterRepository extends RedisDocumentRepository<FictionalCharacter, String> {  
    // Find people by age range  
    Iterable<FictionalCharacter> findByActorAgeBetween(int minAge, int maxAge);  
  
    // Find people by their first and last name  
    Iterable<FictionalCharacter> findByActorFirstNameAndActorLastName(String firstName, String lastName);  
  
    // Draws a circular geofilter around a spot and returns all people in that  
    // radius  
    Iterable<FictionalCharacter> findByActorLocationNear(Point point, Distance distance);  
  
    // Performs full text search on a characters quote  
    Iterable<FictionalCharacter> searchByQuote(String text);  
  
    // Performing a tag search on city  
    Iterable<FictionalCharacter> findByActorAddress_City(String city);  
  
    // Search Characters that have one of multiple skills (OR condition)  
    Iterable<FictionalCharacter> findBySkills(Set<String> skills);  
  
    // Search Characters that have all of the skills (AND condition):  
    Iterable<FictionalCharacter> findBySkillsContainingAll(Set<String> skills);  
}
```

Auto-Complete Out of the Box

`@AutoComplete / @AutoCompletePayload`

```
@Document
public class CharacterEntry {
    @Id private String id;

    @AutoComplete @NonNull private String name;
    @AutoCompletePayload("name") private String type;

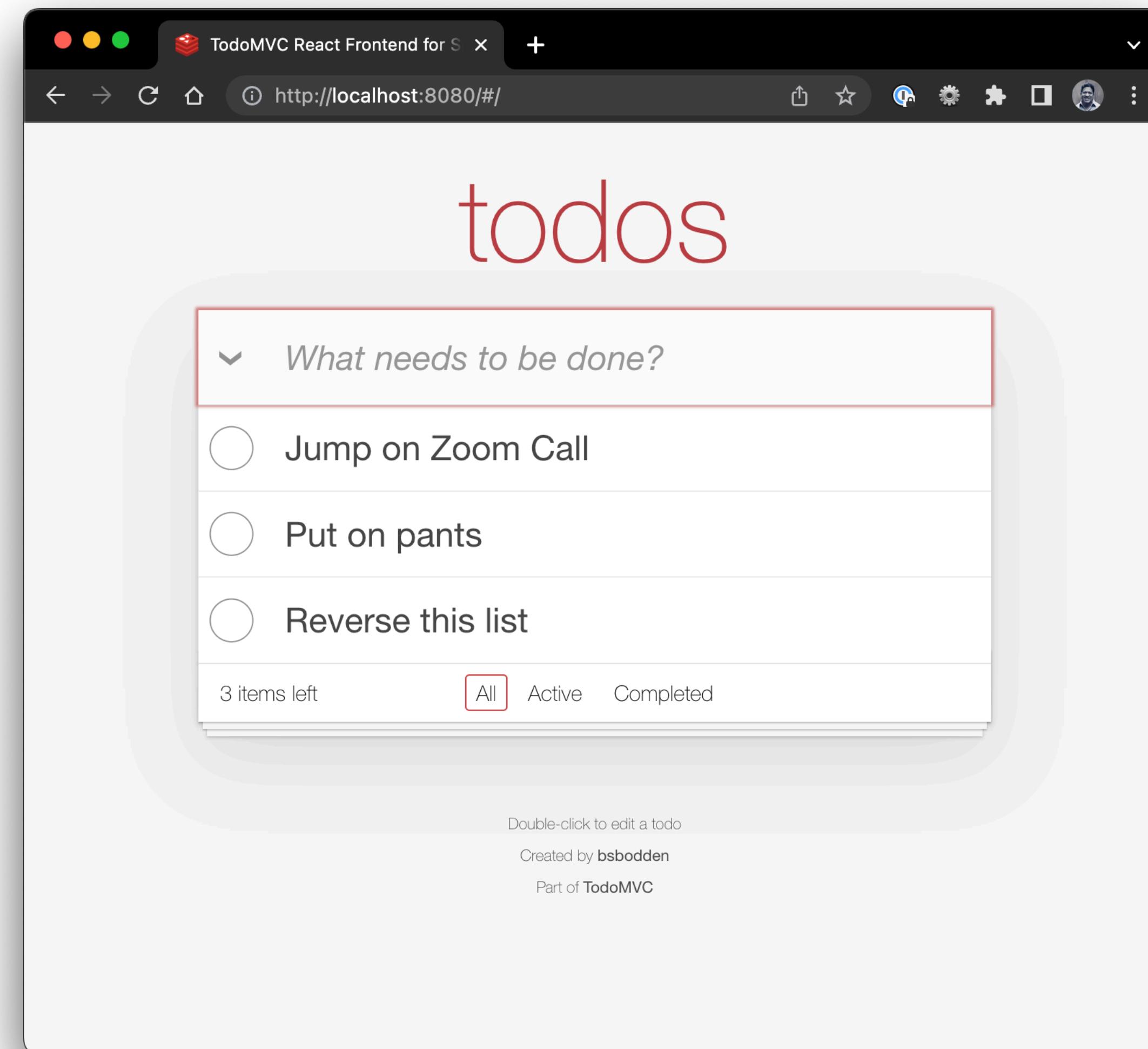
    @SerializedName("first appearance")
    @AutoCompletePayload("name") private String firstAppearance;
    @AutoCompletePayload("name") private Integer appearances;
}
```

```
public interface CharacterEntryRepository
    extends RedisDocumentRepository<CharacterEntry, String> {

    List<Suggestion> autoCompleteName(String query);
    List<Suggestion> autoCompleteName(String query, AutoCompleteOptions options);
}
```

Redis OM Todos

A Basic Redis OM Example



Searching...

High Expectations

There's a growing expectation for effective search functionality

Unstructured Data is "high-dimensional"

Needs to capture "meaning and context" in unstructured data

Encompasses Recommendation Engines and Similarity Search

Search Results for "red tops under 20"

Size

- S (4)
- M (5)
- L (4)

Color

Category

- Tops (5)
- Outerwear (10)
- Bottoms (2)
- Dresses (1)

Red Tops \$0 - \$20 X Red X Tops X Clear All

Sort by: Best Match ▾

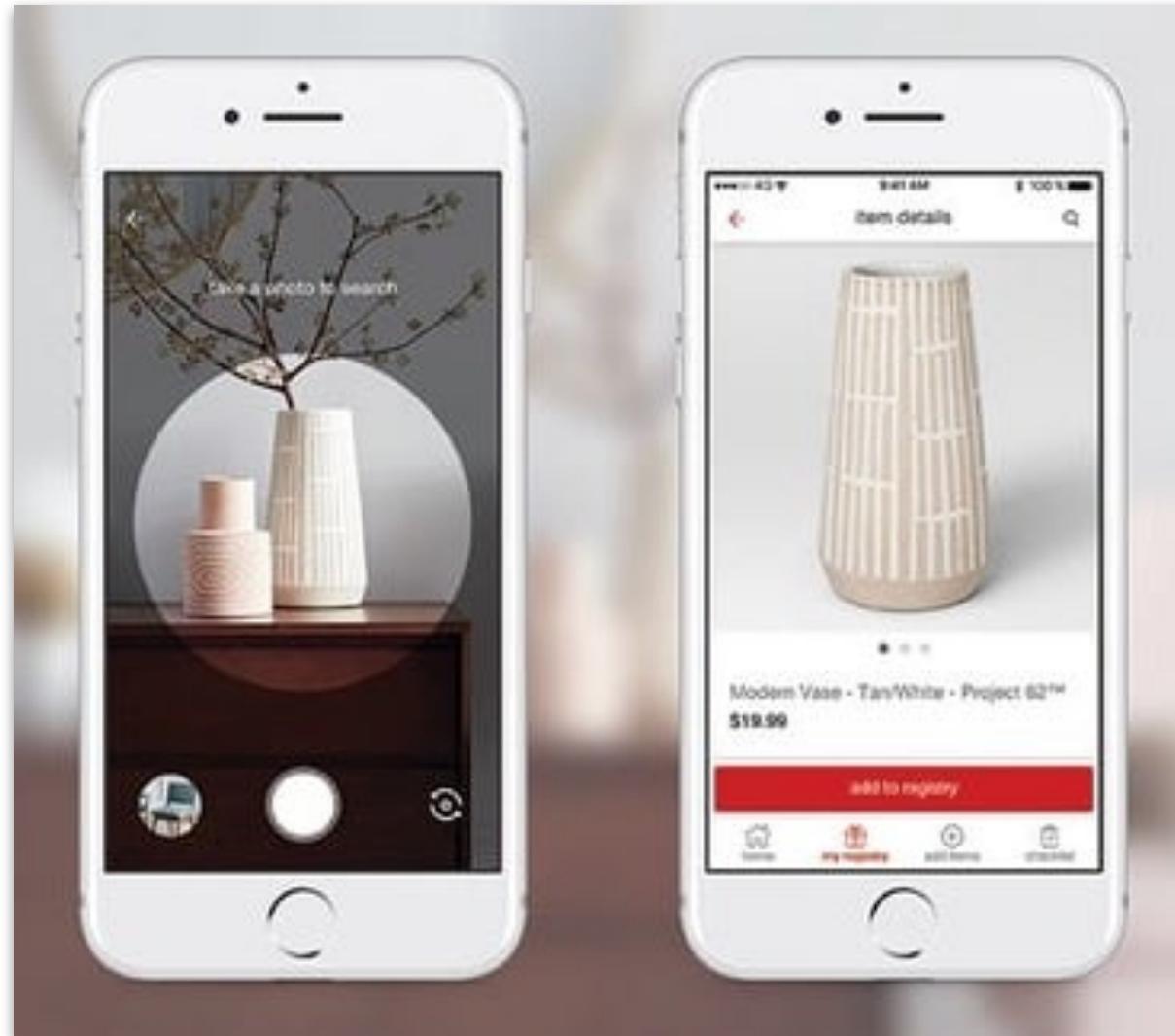
5 results

Take a Drive Striped Blouse - Red
\$34.99 \$19.97
★★★★★ (2)

You're My Everything Rust Red Top
\$24.99 \$9.97
★★★★★ (2)

On The Map Long Sleeve Top - Maroon
\$19.99 \$9.97
★★★★★ (2)

Forever & Always Burgundy Open Back Sweater
\$24.99 \$12.97
★★★★★ (2)



Visual Search

Natural Language Search

Hi, Brian

On Order 0 items

Amazon Prime Try Prime ▾

Gift Card Balance Reload \$100, Get \$5 ▾

Customer Since 2002

Try Amazon Prime today and get unlimited fast, FREE shipping See more

The Joy Clojure

Clojure

Feature Extraction Foundations and Applications

Programming Elixir 1.2

Computational Methods of Feature Selection

CLOJURE FOR THE BRAVE AND TRUE

Recommendations for you in Books

LET OVER LAMBDA

PUBLICIZE YOUR BOOK!

THINKING IN JAVA

ELOQUENT RUBY

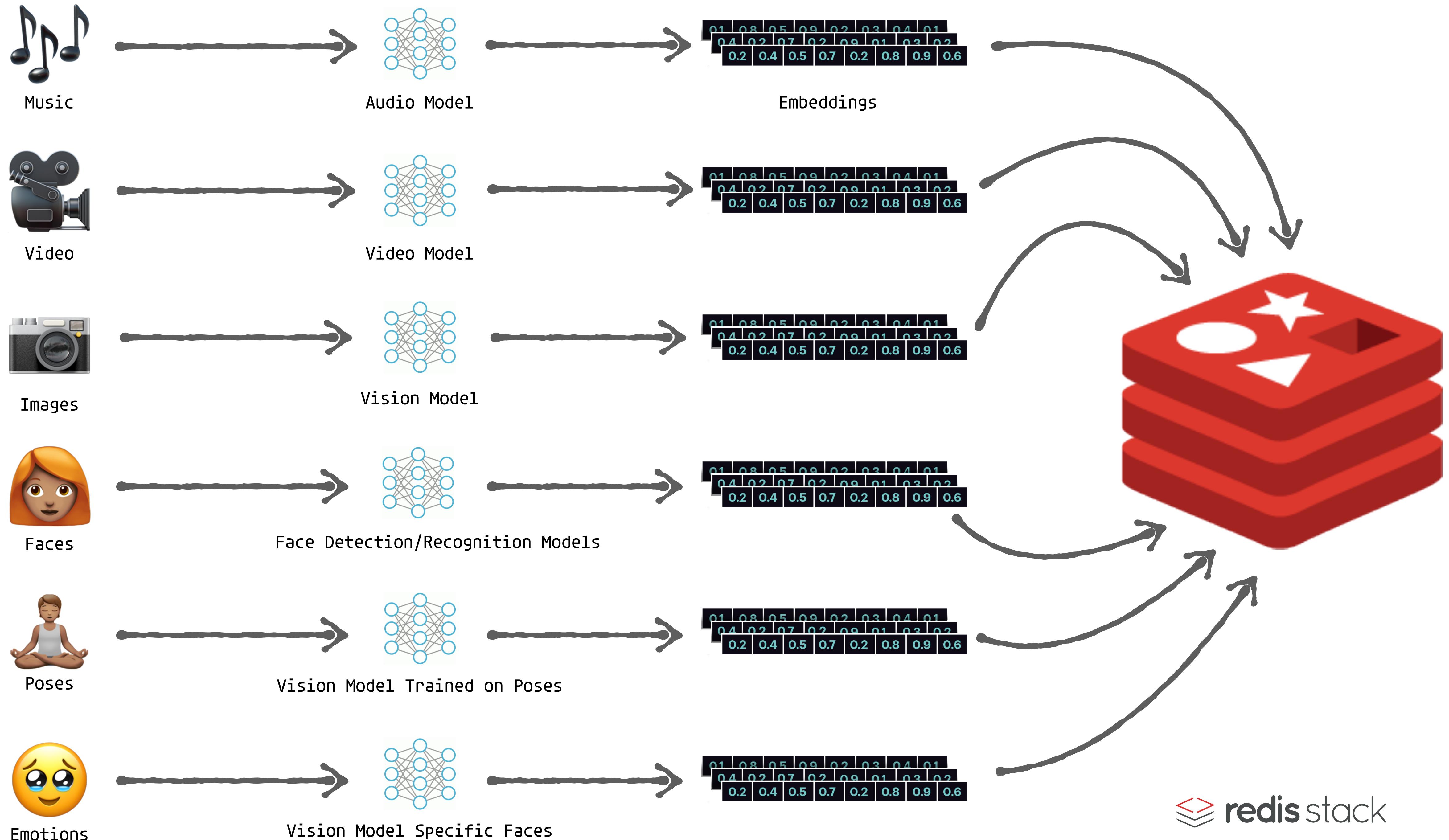
MASTERING TRIANGLE CHOKES

THE SCHEME PROGRAMMING LANGUAGE

Recommenders

Vector Similarity in Redis

Redis as Vector Database



Vector Similarity in Redis

Index and query vector data stored as **BLOBS** in Redis **Hashes/JSON**

3 distance metrics: Euclidean, Internal Product and Cosine

2 indexing methods: HNSW and Flat

Hybrid queries combined with GEO, TAG, TEXT or NUMERIC

Celebrity Match Demo

Facial Similarity Search

Face Detection/Extraction

A peek under the hood



Let's look at the code to detect and extract faces...

Celebrity Match Demo

Breakdown

- 1 A **Celebrity** domain mapped to **Redis Hashes**
- 2 Spring Data **Repositories** powered by **RedisSearch**
- 3 **@Indexed** annotated field for image embeddings
- 4 **@Vectorize** annotated field to generate embeddings
- 5 Upload image, detect/extract faces, gen. input embedding
- 6 Entity **Streams** to query for K nearest neighbors
- 7 Display **results**

eIndexed and eVectorize

TLDR

The source of the embeddings and how to generate them

How to generate the Vector search schema field

```
@Vectorize()  
    destination = "imageEmbedding",  
    embeddingType = EmbeddingType.FACE  
)  
@NonNull  
private String imageResource;  
  
@Indexed()  
    schemaFieldType = SchemaFieldType.VECTOR, //  
    algorithm = VectorAlgo.HNSW, //  
    type = VectorType.FLOAT32, //  
    dimension = 512, //  
    distanceMetric = DistanceMetric.L2, //  
    initialCapacity = 10  
)  
private byte[] imageEmbedding;
```

Redis VSS Celebrities Demo

Redis VSS Celebs

Celebrity Match Demo

A Vector Similarity Search Example powered by Redis Stack.

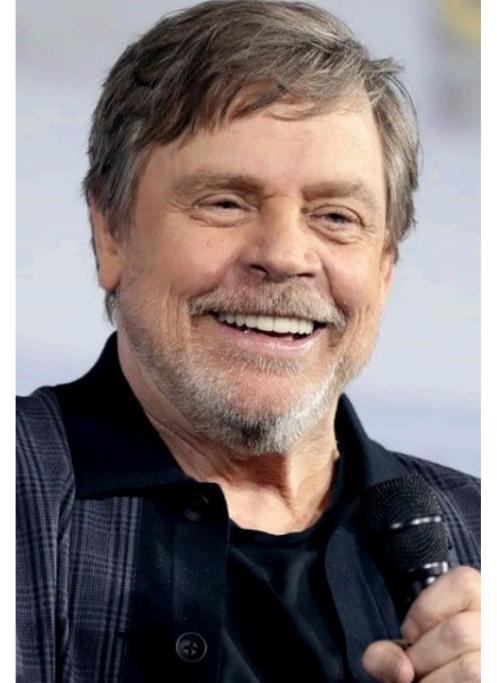
Choose File No file chosen

Upload

New to Redis Stack? [Visit the homepage](#) and read about [Vector Similarity Search](#).

Redis VSS Celebrities Demo

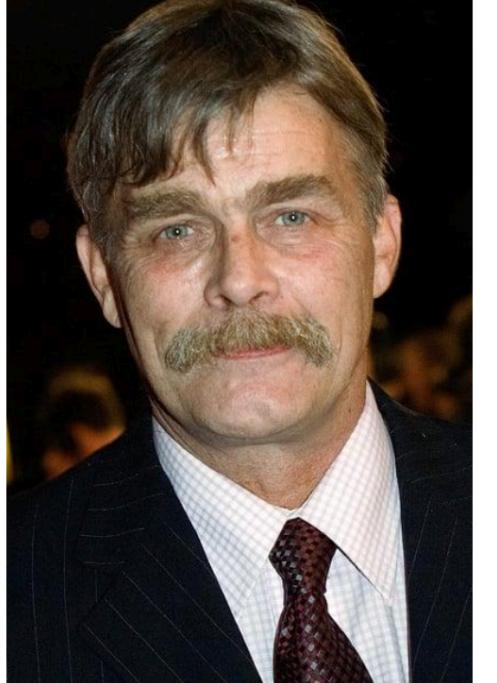
Redis VSS Celebs



Mark Hamill

View

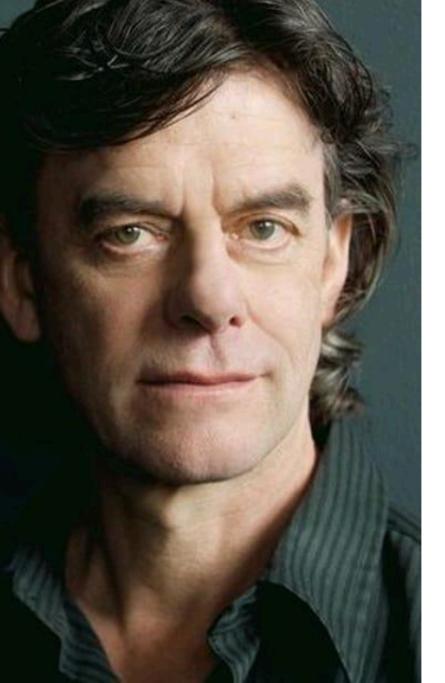
24.564



Nicholas Campbell

View

8.302



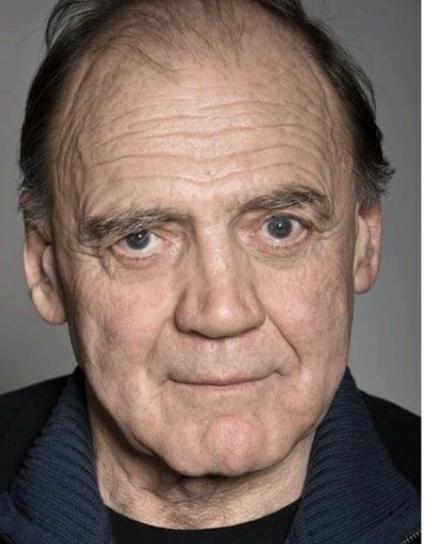
Tom McCamus

View

5.953







Fashion Product Finder

Image & Text - Hybrid Similarity Search

Redis VSS Demo x

http://localhost:8080

Redis Vector Search Demo About ▼ Talk With Us

Fashion Product Finder

This demo uses the built-in Vector Search capabilities of Redis Stack to show how unstructured data, such as images and text, can be used to create powerful search engines.

[Apply Filters](#) [Load More Products](#)

3000 searchable products



Rocky S Women White Handbag

View Similar: [By Text](#) [By Image](#)



Rocky S Women Blue Handbag

View Similar: [By Text](#) [By Image](#)



Rocky S Women Brown Handbag

View Similar: [By Text](#) [By Image](#)



Rocky S Women Red Clutch

View Similar: [By Text](#) [By Image](#)



Puma Unisex White Cap

View Similar: [By Text](#) [By Image](#)



Red backpack



Blue backpack



Yellow backpack



Green backpack



Purple backpack

The tools and techniques to
unlock the value in
Unstructured Data have
evolved greatly...

Databases like Redis and
frameworks like
Redis OM Spring can help!

<https://github.com/bsbodden/roms-vss-celebs>

<https://github.com/redis/redis-om-spring>

<https://redis.io/docs/stack/get-started/tutorials/stack-spring/>

<https://redis.com/blog/rediscover-redis-for-vector-similarity-search/>

<https://github.com/redis/jedis>

Learn more at Redis Developer

<https://developer.redis.com>

The screenshot shows the Redis Developer website homepage with a dark background. At the top left is the "redisdeveloper" logo. Top right features "Get started" and "Try Free" buttons, with a toggle switch between them. A large white title "The Home of Redis Developers" is centered above a subtitle "Made by developers for developers". Below this are three main sections: "Create" (with a database icon), "Develop" (with a code icon), and "Explore" (with a cluster icon). Each section has a brief description and a "Create a database", "Code your application", or "Explore your data" button with a red arrow. To the right of these sections is a large, faint watermark of a Redis cluster graph. At the bottom, there are three footer cards with "Create a database", "Code your application", and "Explore your data" buttons.

redisdeveloper

Get started Try Free

The Home of Redis Developers

Made by developers for developers

Create

Develop

Explore

Create a database →

Code your application →

Explore your data →

Create a database →

Code your application →

Explore your data →

Learn more at Redis University

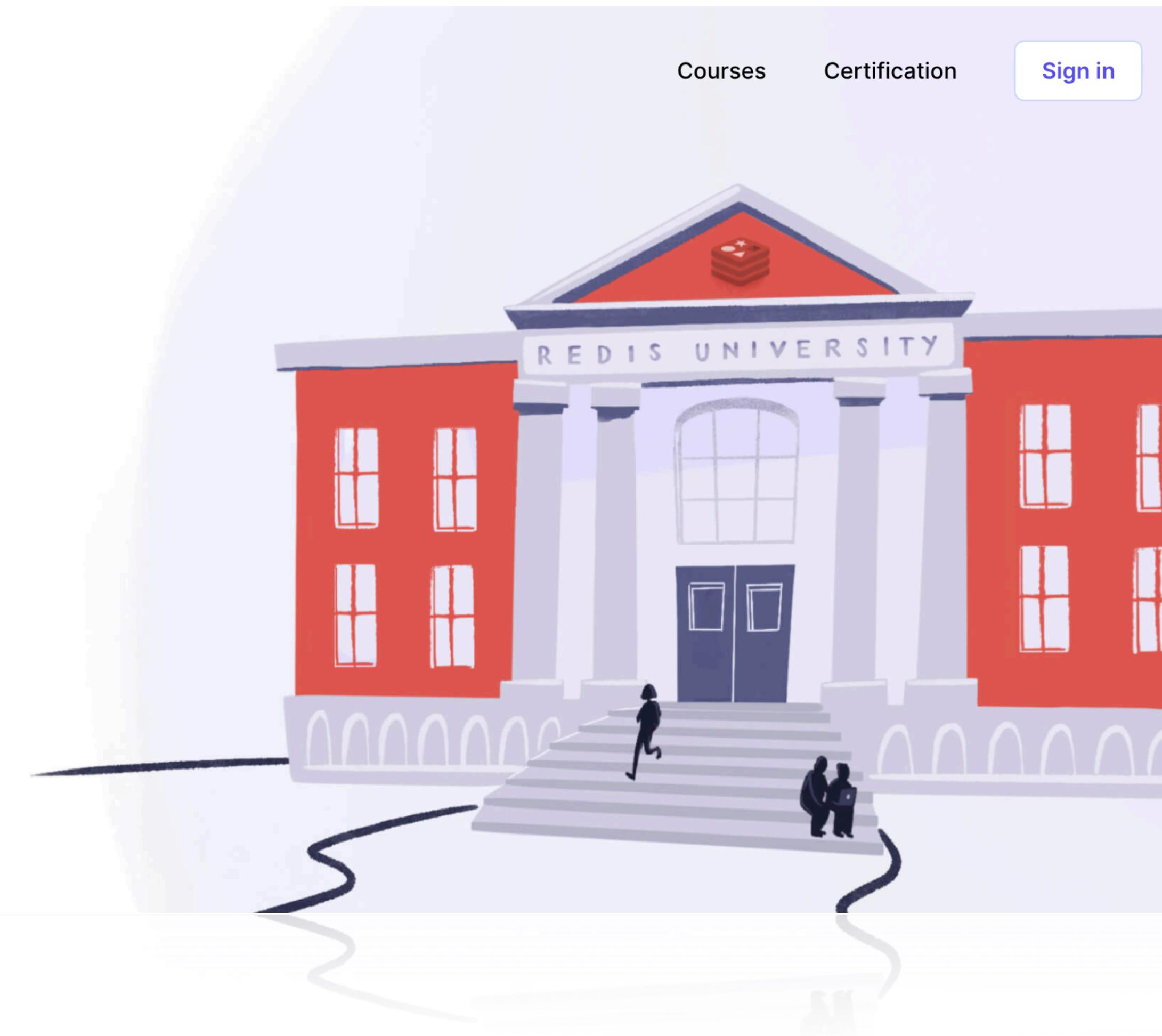
<https://university.redis.com>



Learn Redis at Redis University

Free online courses taught by Redis experts.

Enroll →





Thank you!!!



Thank you!!!