

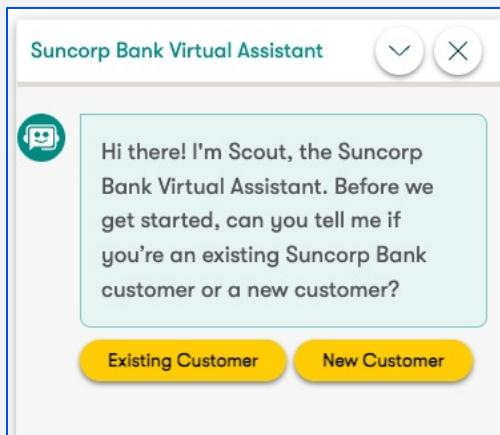
# Embeddings and vector stores

Dr Maria Prokofieva

- Embeddings
- Types of embeddings and uses

# Embeddings

Semantic search  
Recommendation engines  
RAG  
Image similarity and MANY MORE



A screenshot of a Google search results page. The search query is "what should". Below the search bar, a list of suggestions appears, all starting with "what should": "what should i have for dinner", "what should your resting heart rate be", "what should your blood pressure be", "what should i watch", "what should i draw", "what should your heart rate be", "what should resting heart rate be", "what should blood pressure be", "what should my blood pressure be", and "what should i do". To the right of the suggestions, there is a large image of a white cat with blue eyes, identified as a Ragdoll cat. Below the image, there is a snippet of text: "king appearance and gentle". Further down, there is another snippet: "4 most affectionate cat breeds for pet parents who love to snuggle". This snippet includes a small image of a cat and some text about Ragdolls being large, super-friendly cats. At the very bottom of the snippet, there is a link to "Ragdoll Cat Breed | Purina Australia".

# Multimodal embeddings

- **Modalities:** different types of data (text, images, audio, video, medical scans, diagrams, etc.)

A golden retriever puppy playing with a blue ball

The goal of multimodal models: Create a **shared embedding space** where:

- Related things from **any modality** end up close together (e.g., photo of dog + word "dog" + barking sound clip).
- Unrelated things stay far apart.



## Contrastive learning:

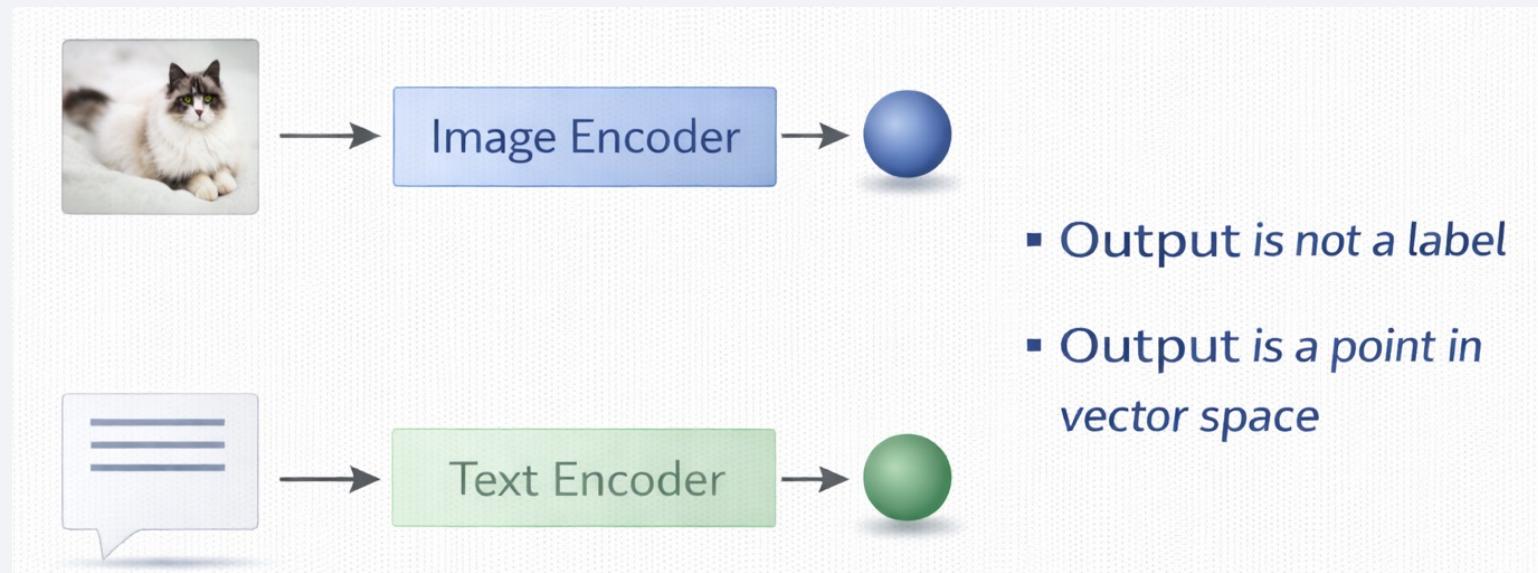
technique that teaches models to

- identify similarities and differences in data
- by grouping similar examples (positive pairs) together in a vector space while
- pushing dissimilar ones (negative pairs) apart.

A fluffy cat lying down with tail wrapped around

# Multimodal embeddings: contrastive learning

- Learn embeddings by pulling matching pairs together
- Pushing non-matching pairs apart
- Distance in vector space is trained to reflect semantic correspondence.



# Contrastive Learning

## Positive pairs

- Represent the **same** underlying concept
- Should be **close** in embedding space

## Negative pairs

- Represent **different** concepts
- Should be **far apart**

Intra-modality contrastive learning:  
Learning structure within the same modality

*A fluffy Ragdoll cat lying down with tail wrapped around*



## Positive intra-modality

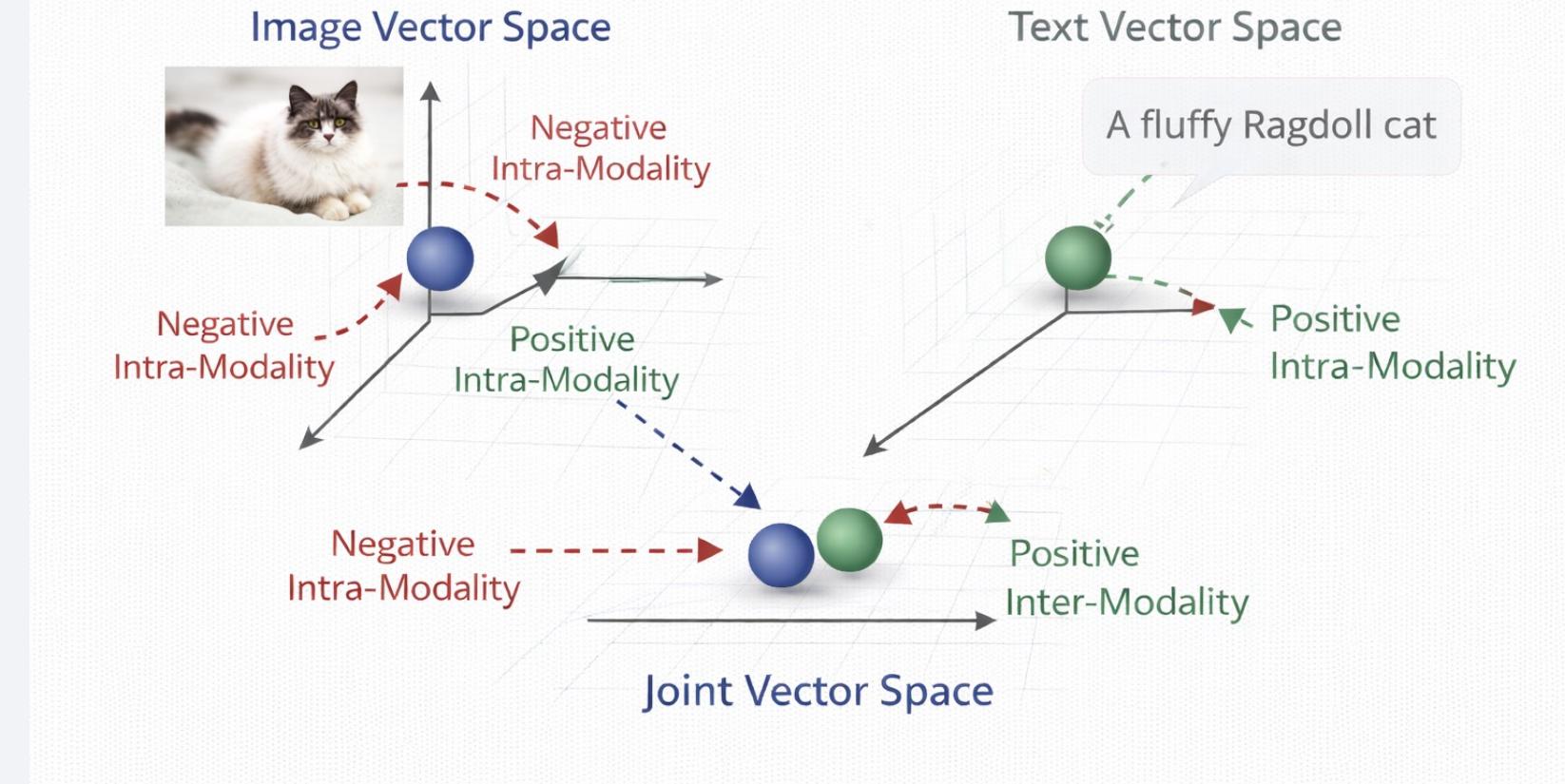
- Image  $\leftrightarrow$  image (same object, different views)
- Text  $\leftrightarrow$  text (paraphrases)

## Negative intra-modality

- Image  $\leftrightarrow$  image (different objects)
- Text  $\leftrightarrow$  text (unrelated meanings)

# Multimodal Vector Space

Combining image and text into a joint vector space



	Positive (pull together)	Negative (push apart)
Intra-modality	Same meaning, same modality	Different meaning, same modality
Inter-modality	Same meaning, different modalities	Different meaning, different modalities

# Contrastive Learning

# Measuring Quality

- There is no universal “good embedding” – quality is task-dependent.
- An embedding is good if geometric relationships in vector space align with the task-relevant notion of similarity.

- Same vocabulary
- Same inputs
- Different geometry

Because the objective changes

- There is no single “true” semantic space.

Cold

biomedical:  
near "influenza",  
"rhinovirus", "hypothermia"

News

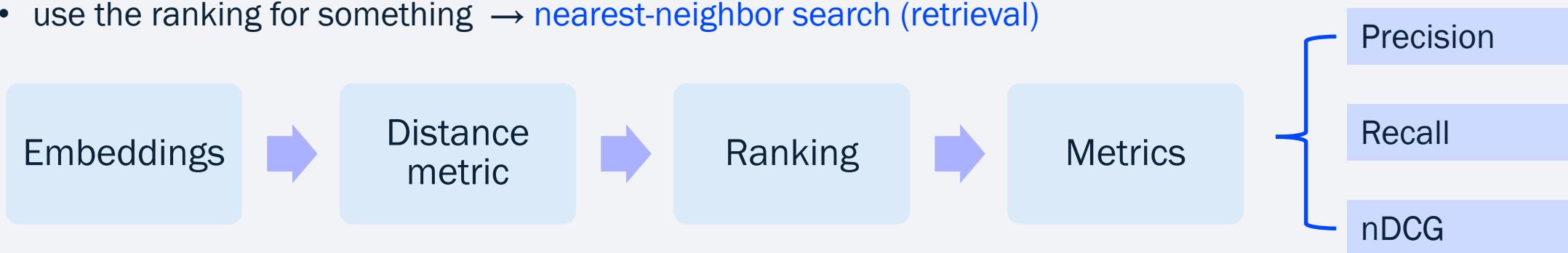
legal:  
near "judgment",  
"precedent", "litigation"

general:  
near "box", "example",  
"situation"

# How Embeddings Turn into Retrieval Metrics

A vector has **no meaning** until you:

- compare it to other vectors
- rank those comparisons
- use the ranking for something → **nearest-neighbor search (retrieval)**



Embeddings are not scored. Retrieval behaviour induced by embeddings is scored.

Did I return junk? → Precision  
Did I miss good stuff? → Recall  
Did I order the good stuff correctly? → nDCG