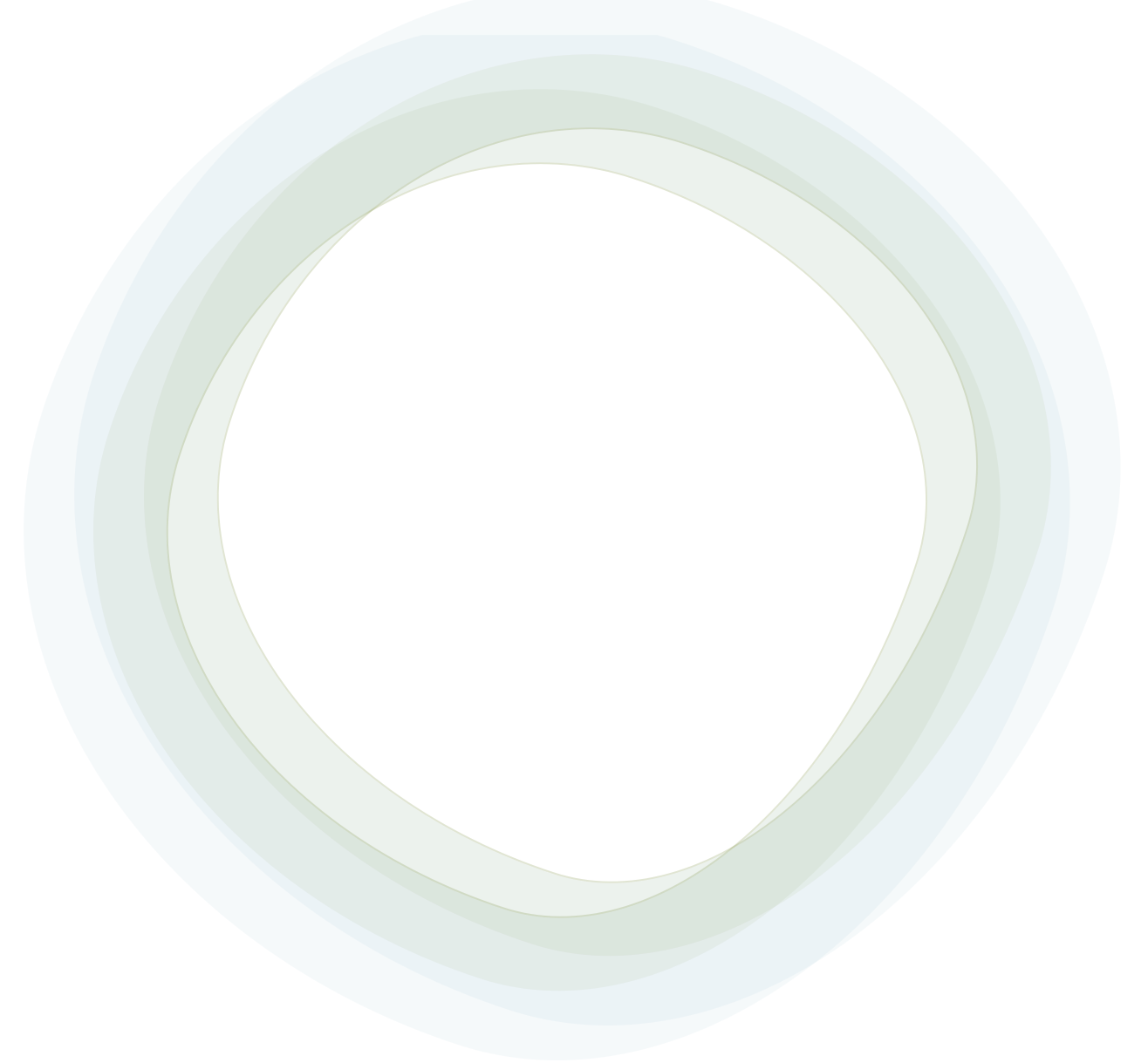


Multimodal LLMs & VLA

How machines get from seeing and hearing to doing

JORDAN HILL UWA · 4–6 PM



Which is harder?

REQUEST A

Build me a gravitational model of the Solar System and explain general relativity.

*first step: find a **book** on general relativity*

REQUEST B

Could you make me a cup of tea?

*first step: the **red mug** — top shelf, above the kettle*

Which is harder?

REQUEST A

Build me a gravitational model of the Solar System and explain general relativity.

*first step: find a **book** on general relativity*

*need to know: the field equations — **written down**, searchable*

REQUEST B

Could you make me a cup of tea?

*first step: the **red mug** — top shelf, above the kettle*

*need to know: where the mug is **right now** — nobody wrote it down*

Which is harder?

REQUEST A

Build me a gravitational model of the Solar System and explain general relativity.

*first step: find a **book** on general relativity*

*need to know: the field equations — **written down**, searchable*

*actions: write the code, run it, fix it — every retry **free***

REQUEST B

Could you make me a cup of tea?

*first step: the **red mug** — top shelf, above the kettle*

*need to know: where the mug is **right now** — nobody wrote it down*

*actions: reach, grasp, pour — each changes the room; **none undo***

Which is harder?

REQUEST A

Build me a gravitational model of the Solar System and explain general relativity.

*first step: find a **book** on general relativity*

*need to know: the field equations — **written down**, searchable*

*actions: write the code, run it, fix it — every retry **free***

REQUEST B

Could you make me a cup of tea?

*first step: the **red mug** — top shelf, above the kettle*

*need to know: where the mug is **right now** — nobody wrote it down*

*actions: reach, grasp, pour — each changes the room; **none undo***

Which is harder?

Which is harder?

REQUEST A

Build me a gravitational model of the Solar System and explain general relativity.

*first step: find a **book** on general relativity*

*need to know: the field equations — **written down**, searchable*

*actions: write the code, run it, fix it — every retry **free***

REQUEST B

Could you make me a cup of tea?

*first step: the **red mug** — top shelf, above the kettle*

*need to know: where the mug is **right now** — nobody wrote it down*

*actions: reach, grasp, pour — each changes the room; **none undo***

Which is harder?
Difficulty is a bit relative.

Hans Moravec noticed in 1988 that

“ it is comparatively easy to make computers exhibit adult level performance on intelligence tests or playing checkers, and **difficult or impossible** to give them the skills of a one-year-old when it comes to perception and mobility ”

Hans Moravec · *Mind Children* · 1988

One task never leaves the text

One task never leaves the text

SOLAR SYSTEM MODEL

Text in, code out, run it, check it, fix it.

Representations all the way through. Cheap retries. Copy-paste the universe.

A CUP OF TEA — STARTING WITH THE MUG

The sentence doesn't contain the scene.

Which shelf? Where's the handle? How heavy? What happens if I knock the kettle?

One task never leaves the text

SOLAR SYSTEM MODEL

Text in, code out, run it, check it, fix it.

Representations all the way through. Cheap retries. Copy-paste the universe.

Why has intelligence in representations advanced faster than intelligence in action?

A CUP OF TEA — STARTING WITH THE MUG

The sentence doesn't contain the scene.

Which shelf? Where's the handle? How heavy? What happens if I knock the kettle?

Agenda

01 Self-supervision

using raw data as its own supervisor

03 Signals into the language model

how text-predictors become multi-modal

05 World models

what changes when you predict before you act

02 Joint embedding

building shared latent representations

04 Vision–language–action

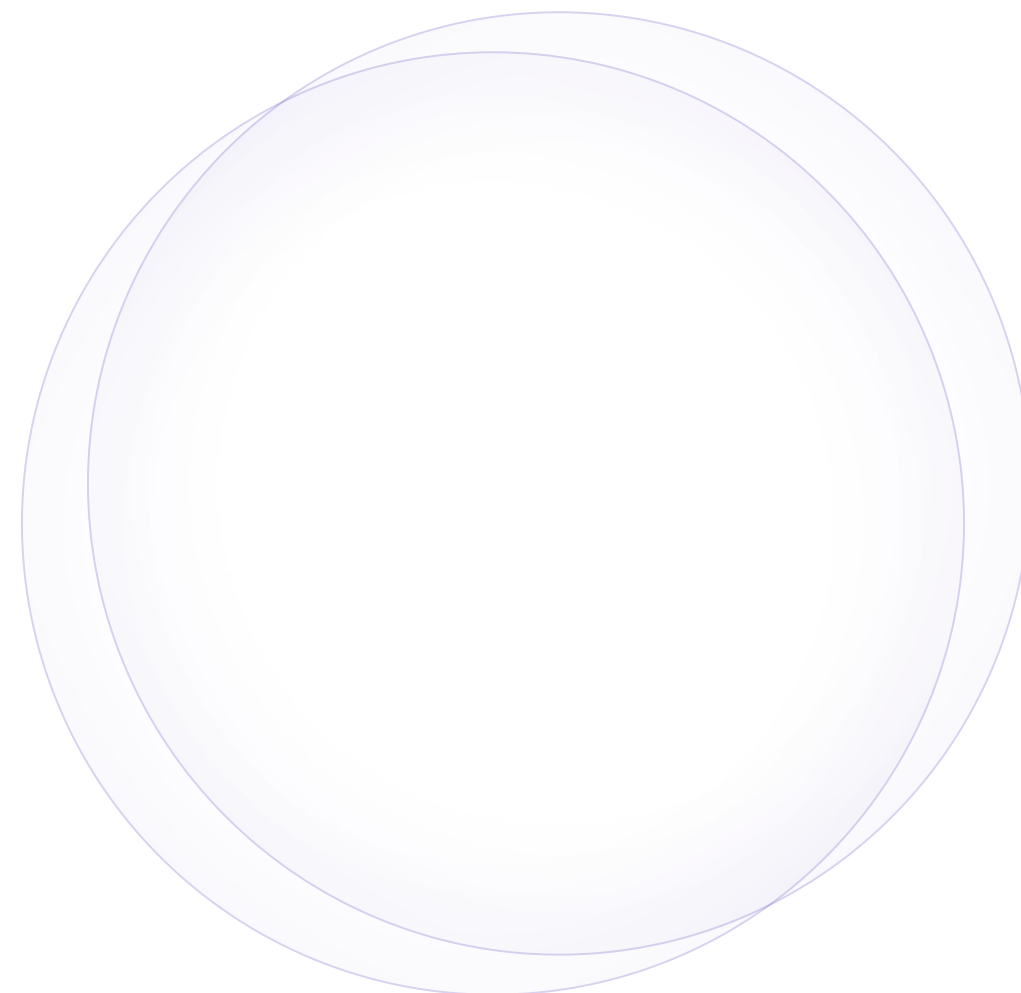
learning actions from examples

predict the next word → relate signals → answer → act → predict consequences

Learning from the next word

Self-supervision in language models

TEXT → HIDDEN TARGETS → A MODEL OF WHAT COMES NEXT

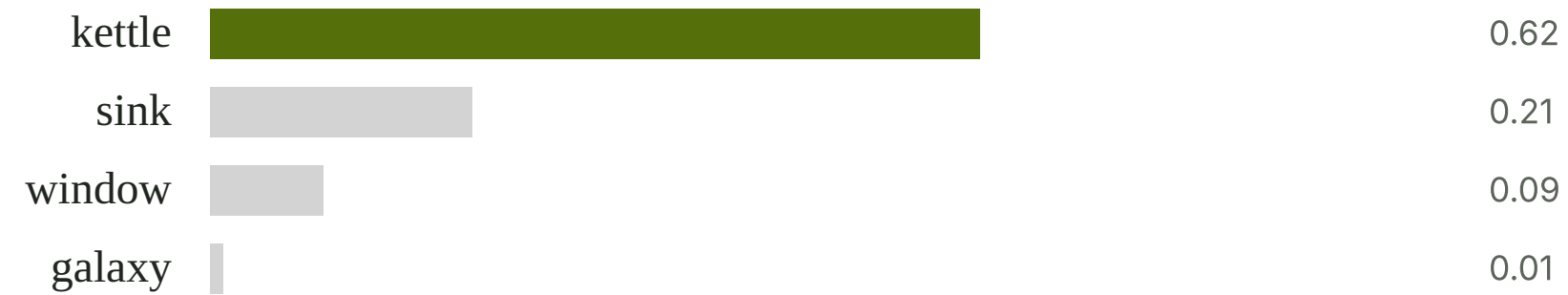


Next-word prediction

"The red mug is above the ."

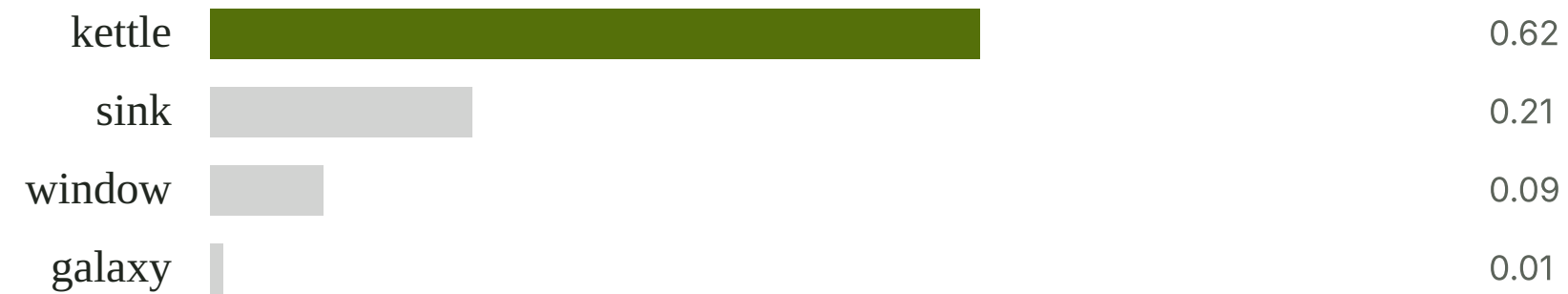
Next-word prediction

"The red mug is above the ."



Next-word prediction

"The red mug is above the ."



No labels: the target is the next word of the text itself.

One sentence, many training examples

CONTEXT	TARGET	GUESS	LOSS
The red mug	mug		

One sentence, many training examples

CONTEXT		TARGET	GUESS	LOSS
The red mug	→	mug		
The red mug	→	is		

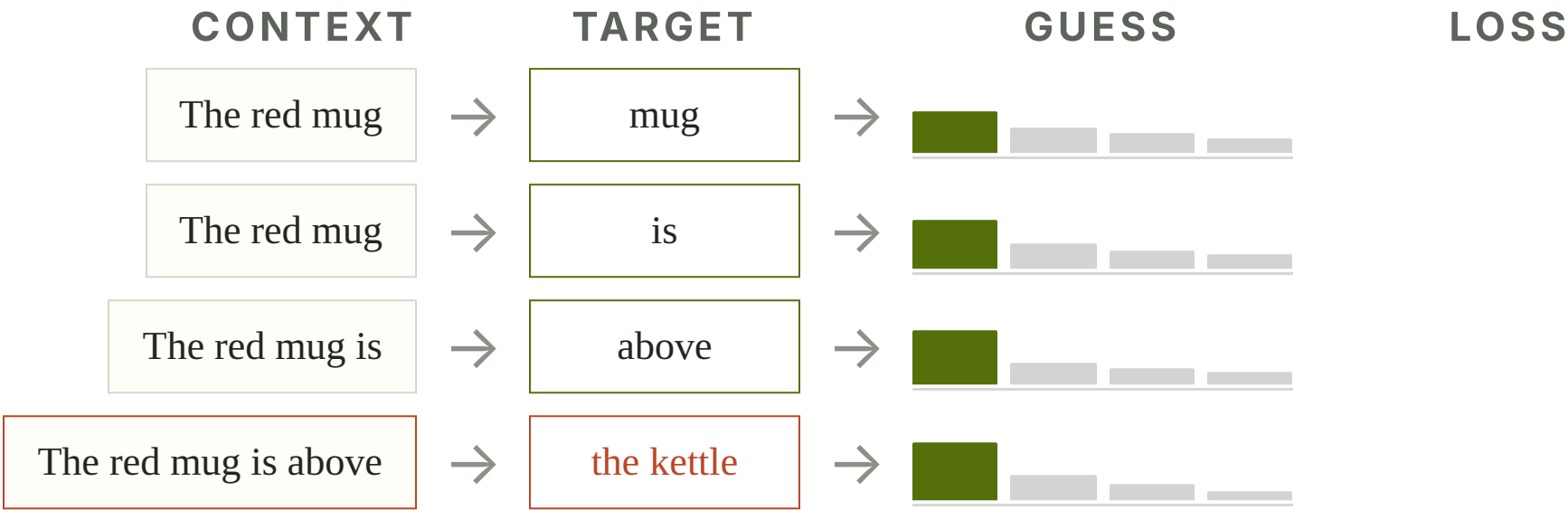
One sentence, many training examples

CONTEXT		TARGET	GUESS	LOSS
The red mug	→	mug		
The red mug	→	is		
The red mug is	→	above		

One sentence, many training examples

CONTEXT		TARGET	GUESS	LOSS
The red mug	→	mug		
The red mug	→	is		
The red mug is	→	above		
The red mug is above	→	the kettle		

One sentence, many training examples



One sentence, many training examples

CONTEXT		TARGET		GUESS		LOSS
The red mug	→	mug	→		→	0.89
The red mug	→	is	→		→	0.69
The red mug is	→	above	→		→	0.54
The red mug is above	→	the kettle	→		→	0.48

One sentence, many training examples

CONTEXT		TARGET		GUESS		LOSS
The red mug	→	mug	→		→	0.89
The red mug	→	is	→		→	0.69
The red mug is	→	above	→		→	0.54
The red mug is above	→	the kettle	→		→	0.48

Each position gives a context, a target and a loss.

Inside the language model: tokens, attention, loss

TOKENS → VECTORS

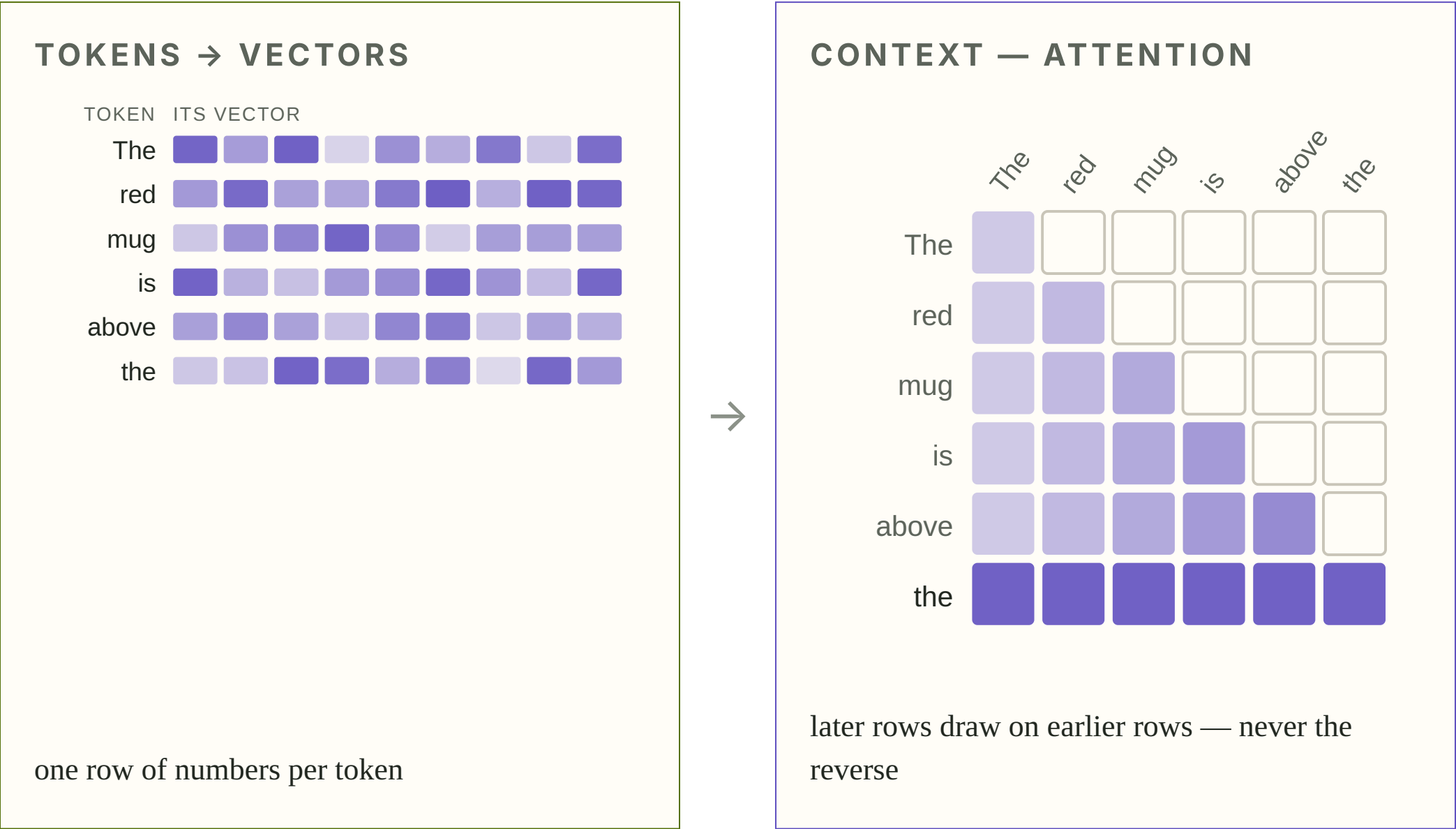
TOKEN	ITS VECTOR
The	
red	
mug	
is	
above	
the	

one row of numbers per token

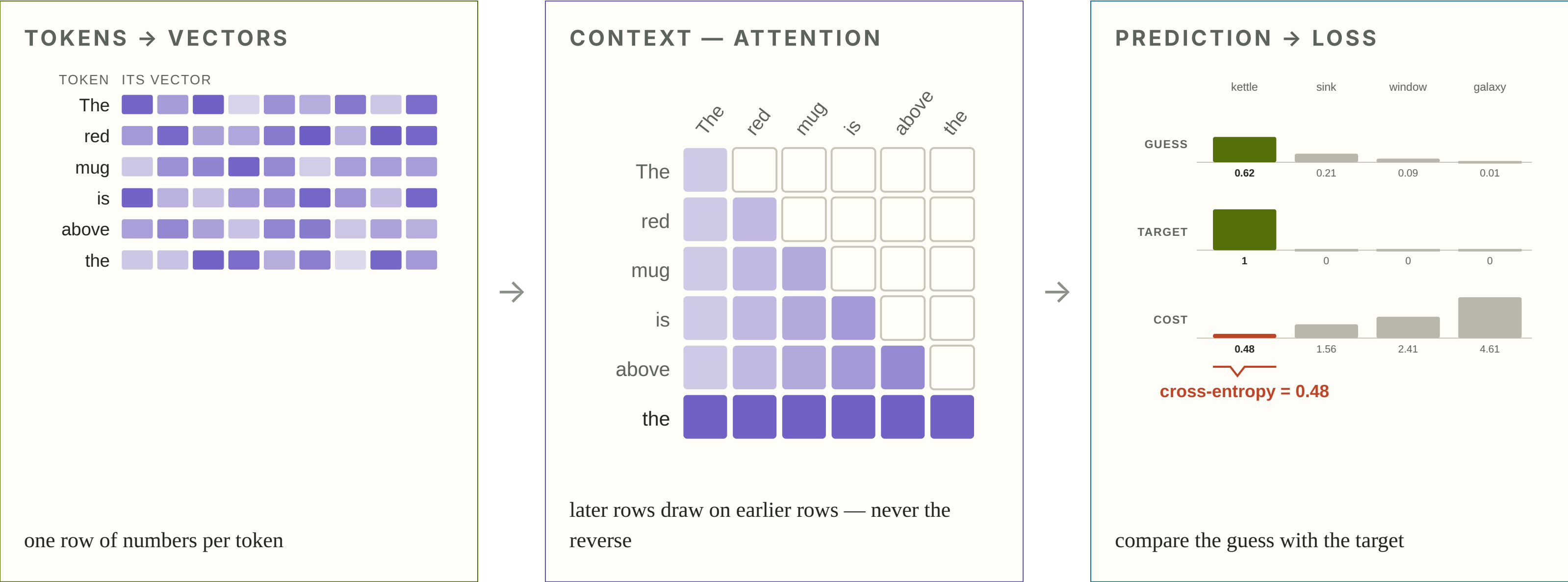
→

→

Inside the language model: tokens, attention, loss



Inside the language model: tokens, attention, loss



Self-supervised learning

THE RECIPE

Hide part of the data. Predict it from the rest. Let the data mark the answer.

Self-supervised learning

THE RECIPE

Hide part of the data. Predict it from the rest. Let the data mark the answer.

WHY IT SCALES

Existing text supplies targets without hand-written labels.

Self-supervised learning

THE RECIPE Hide part of the data. Predict it from the rest. Let the data mark the answer.

WHY IT SCALES Existing text supplies targets without hand-written labels.

WHAT IT EXPLAINS The request is richly represented in text, mathematics and code.

Self-supervised learning

THE RECIPE Hide part of the data. Predict it from the rest. Let the data mark the answer.

WHY IT SCALES Existing text supplies targets without hand-written labels.

WHAT IT EXPLAINS The request is richly represented in text, mathematics and code.

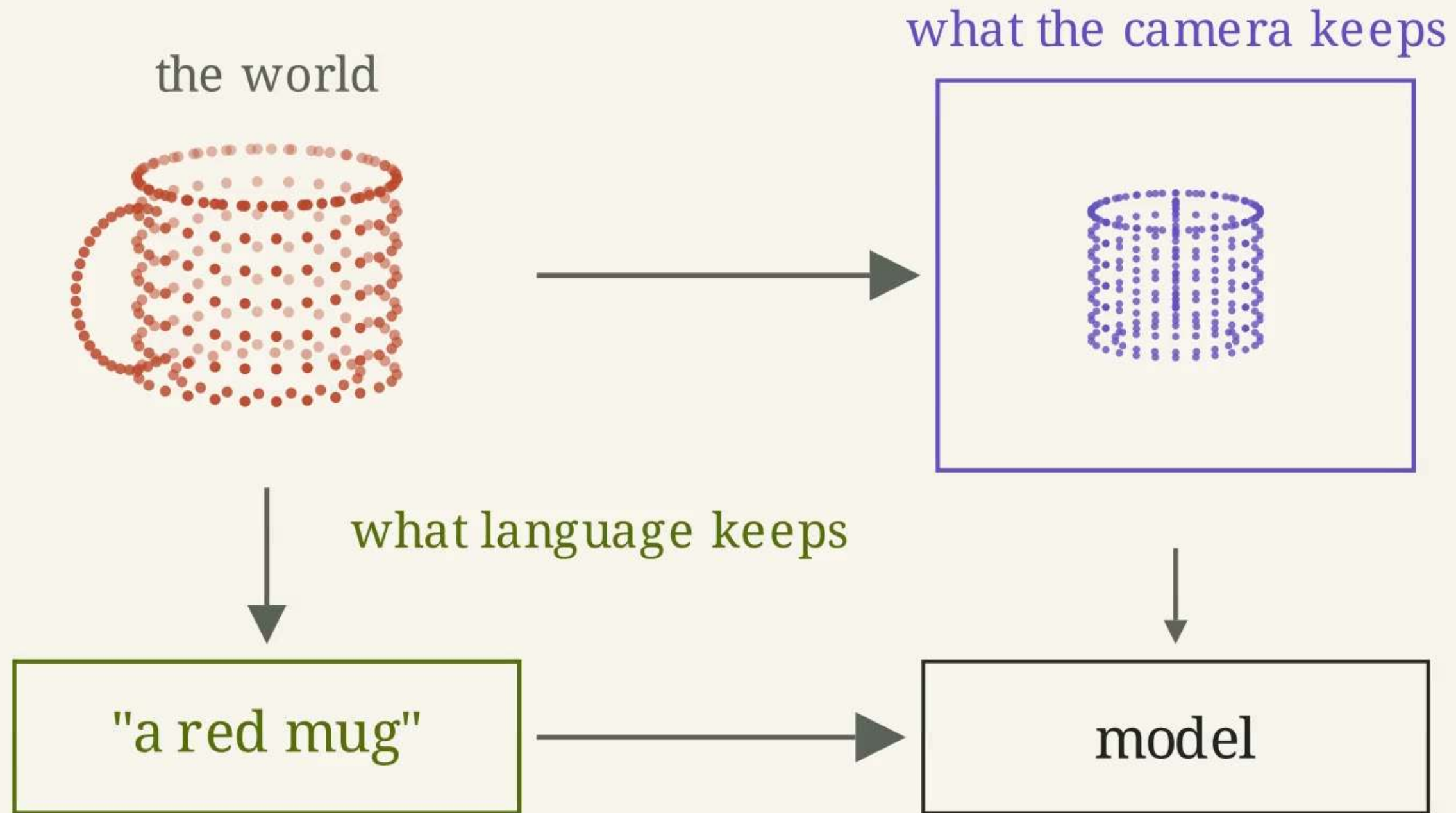
Text supervises text because nearby words are related.

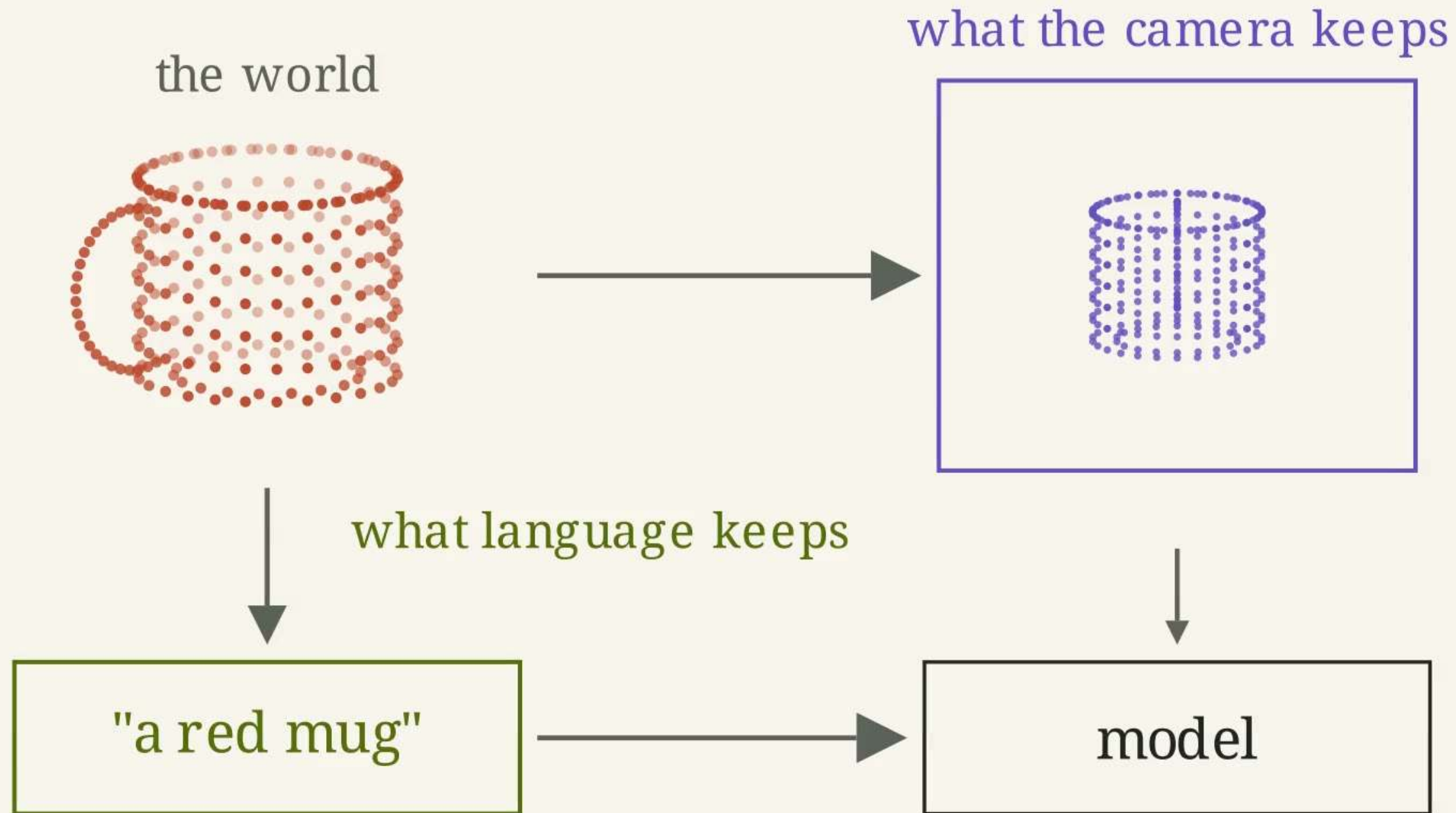
What else arrives already paired?

A modality is a projection of the world



Each modality carries information the others do not.



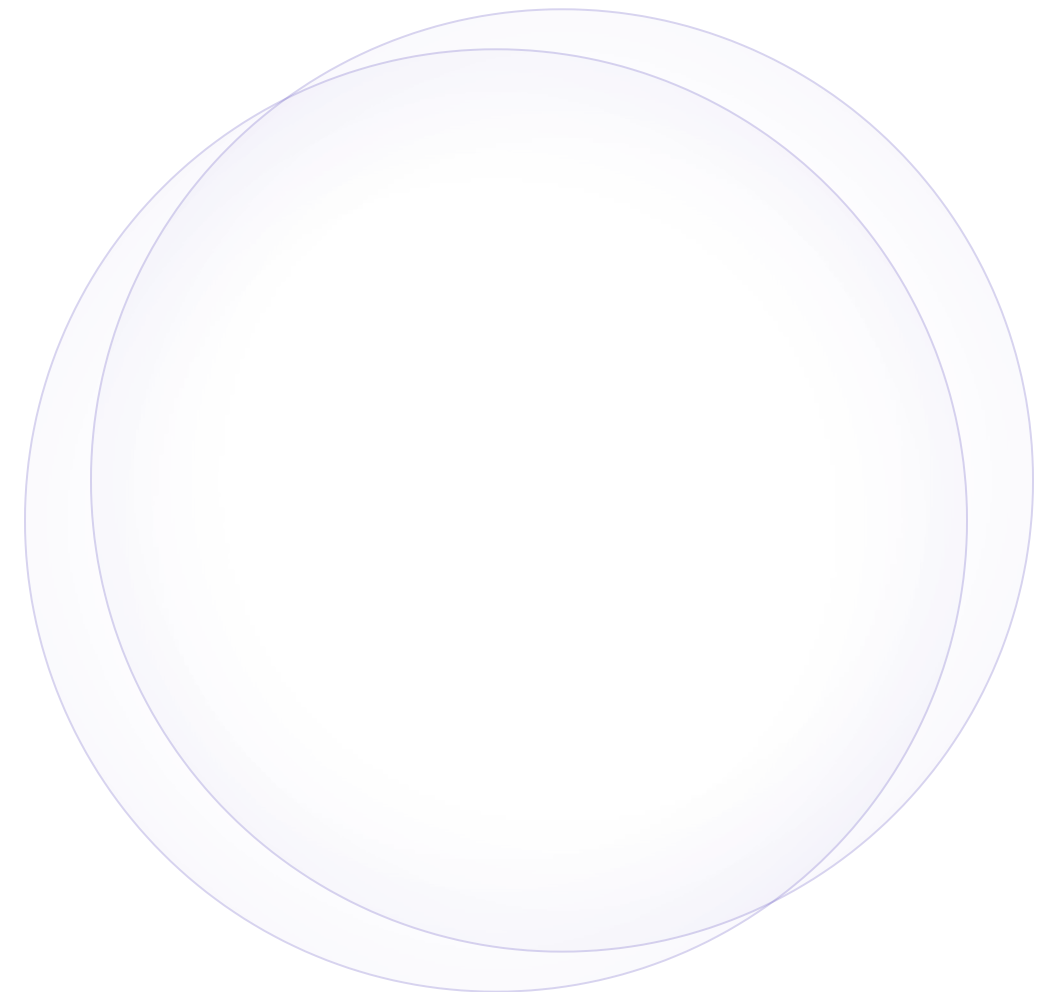


One world produces many signals.
No single signal contains the whole world.

Joint embedding

Learning one space for two signals

PAIRED DATA → ENCODERS → SIMILARITY → MATCHING LOSS → REPRESENTATION



Paired data supplies the target



CAPTION FOUND BESIDE THE IMAGE

"a red cup beside a blue cup in the sink"



TRAINING TARGET

Which caption belongs with this image?
the pairing is the answer

Token identity and position are added into **one vector**.

WHAT THE TOKEN IS

"mug" → embedding table

the same vector wherever it appears

+

WHERE IT SITS

position 3 → positional embedding

the same vector whatever the word

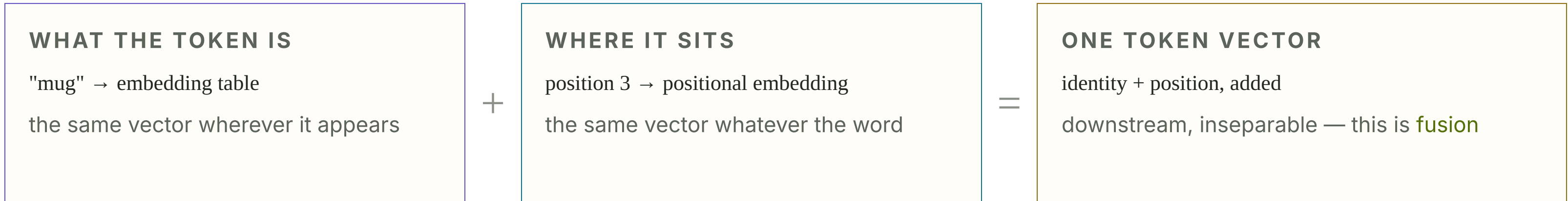
=

ONE TOKEN VECTOR

identity + position, added

downstream, inseparable — this is **fusion**

Token identity and position are added into **one vector**.



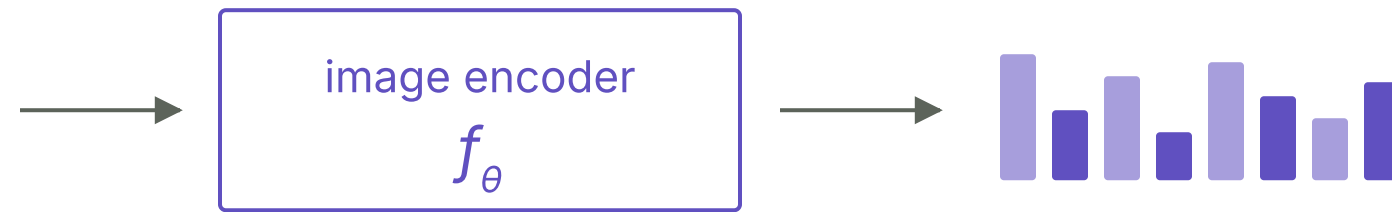
Combining sources into one representation is **fusion**. Any signal can be encoded on its own.

How do two separate encoders learn one joint representation?

Two encoders, one coordinate system



the image — pixels

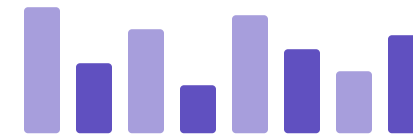
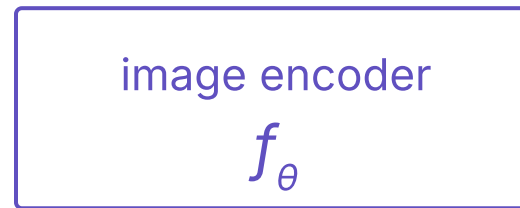


V one vector for the whole image
d numbers, computed from the pixels

Two encoders, one coordinate system



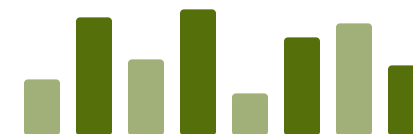
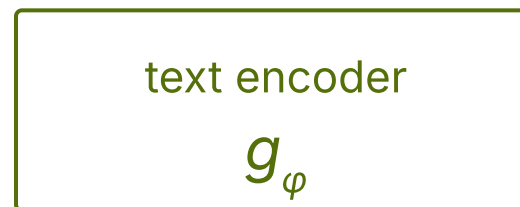
the image — pixels



V one vector for the whole image
d numbers, computed from the pixels

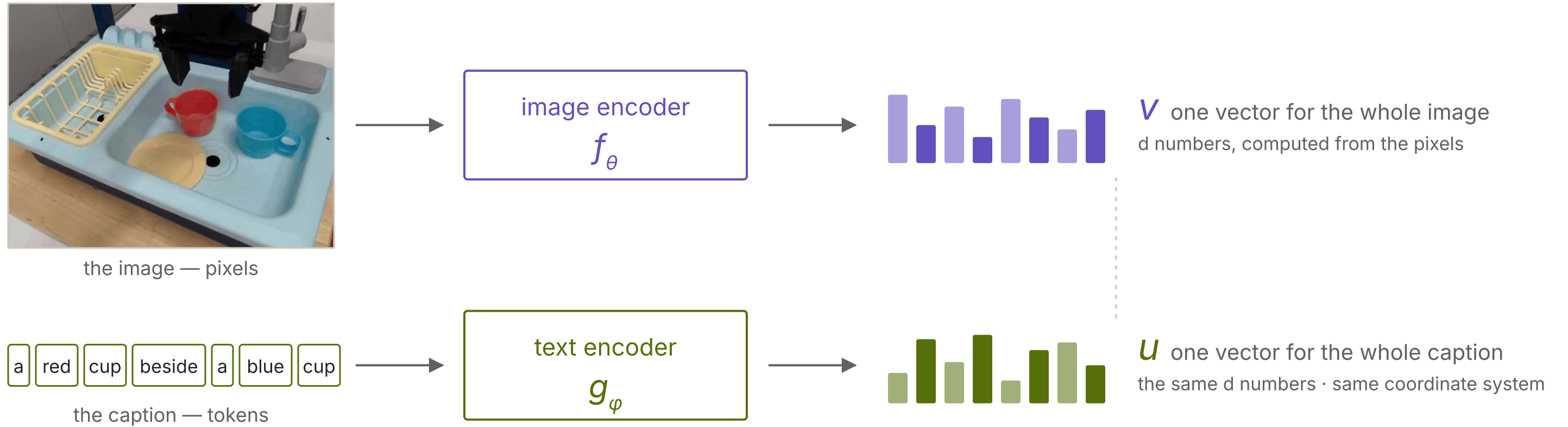
a red cup beside a blue cup

the caption — tokens



U one vector for the whole caption
the same d numbers · same coordinate system

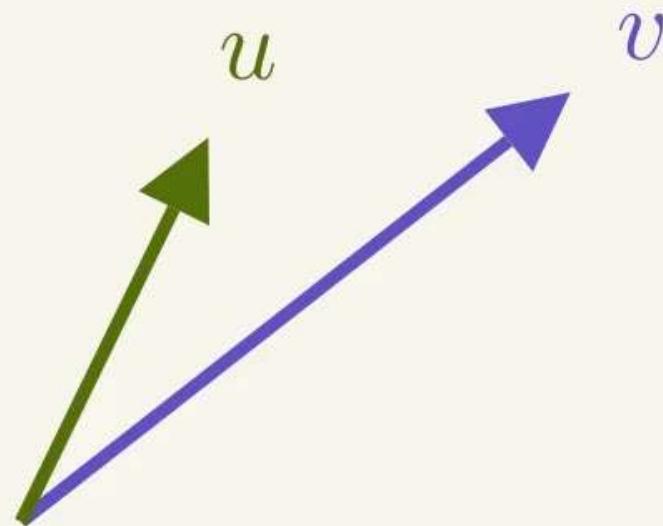
Two encoders, one coordinate system



$$s_{ij} = \hat{v}_i \cdot \hat{u}_j$$

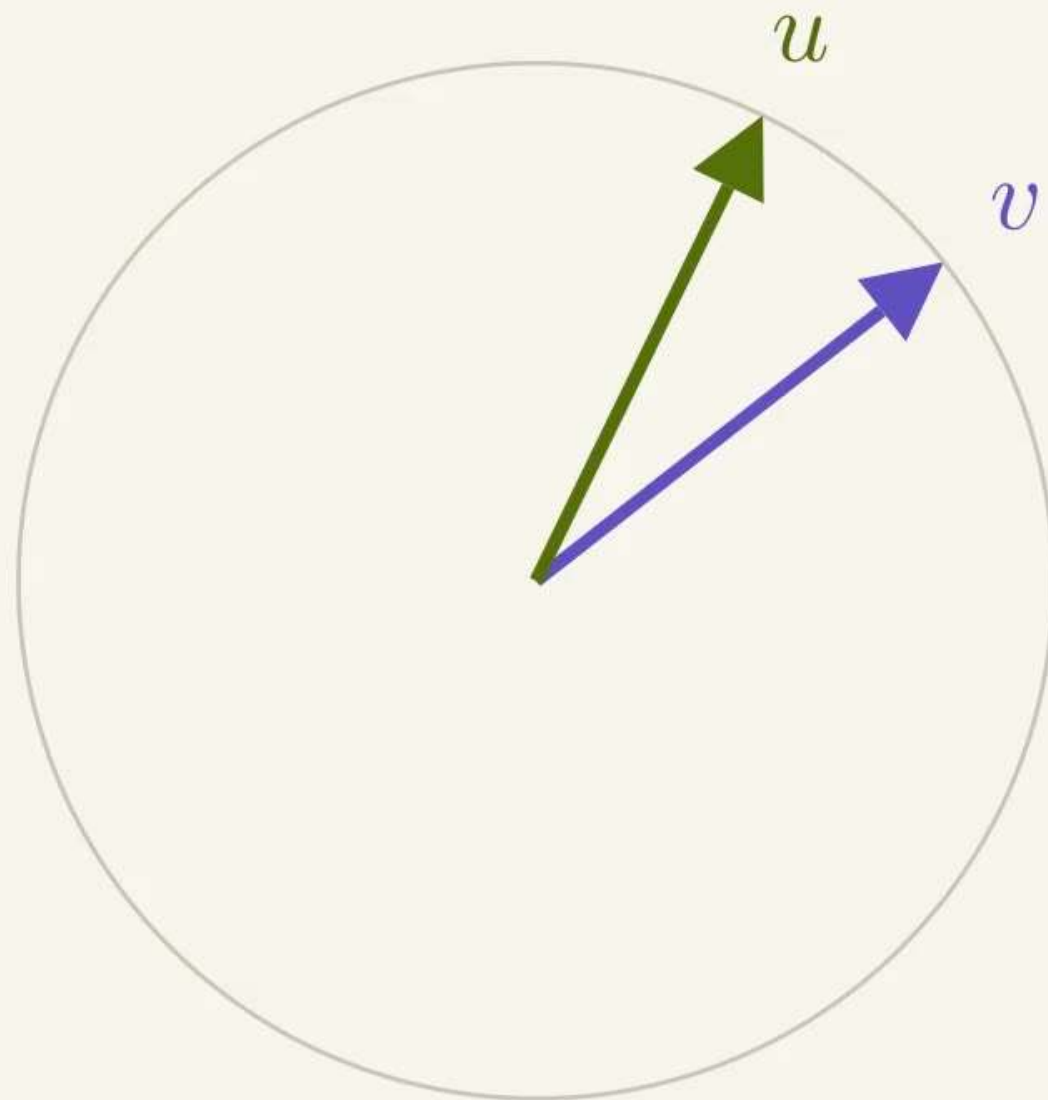
one candidate pairing: normalise each vector once, take the dot product — one scalar in $[-1, 1]$

normalise, then dot product



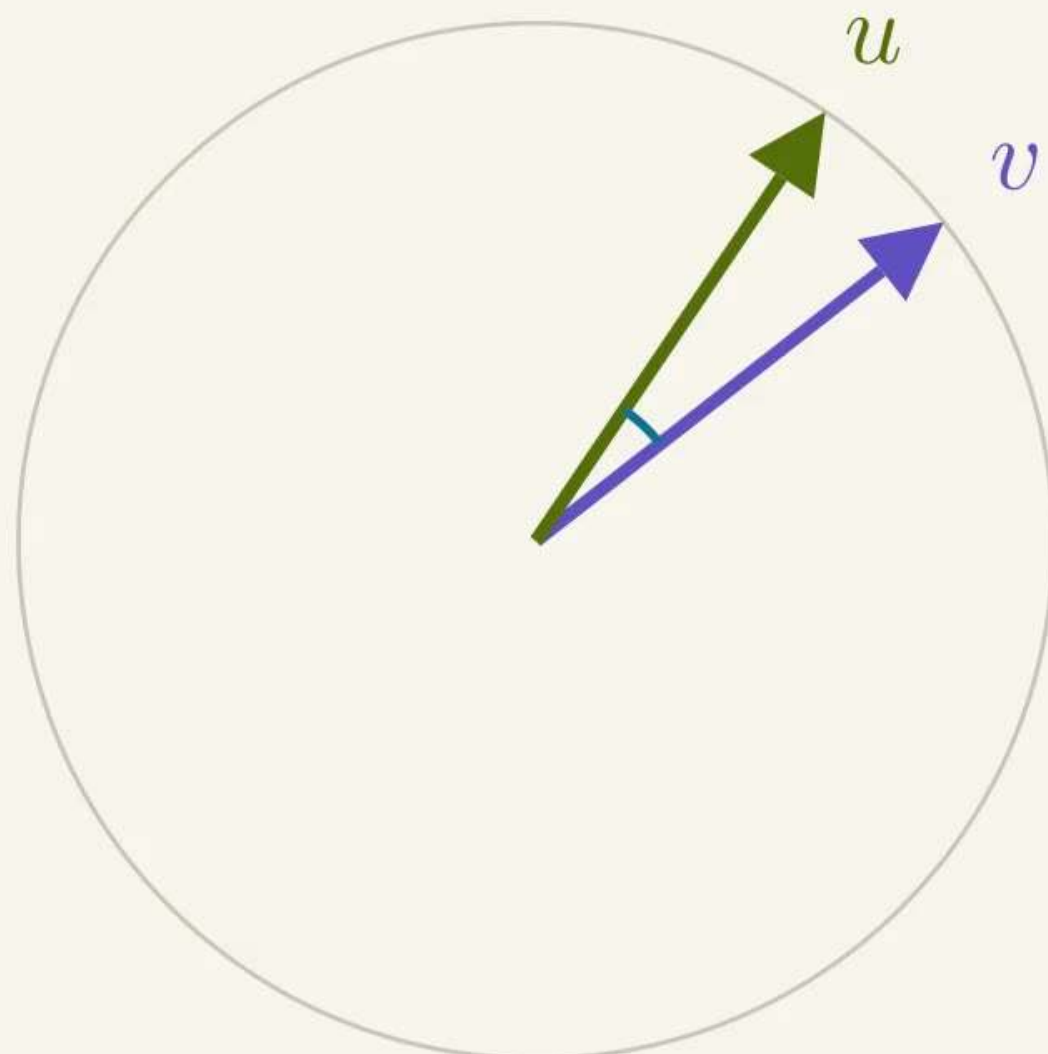
an image vector and a caption vector — different lengths

normalise, then dot product



normalise: keep the direction, forget the length

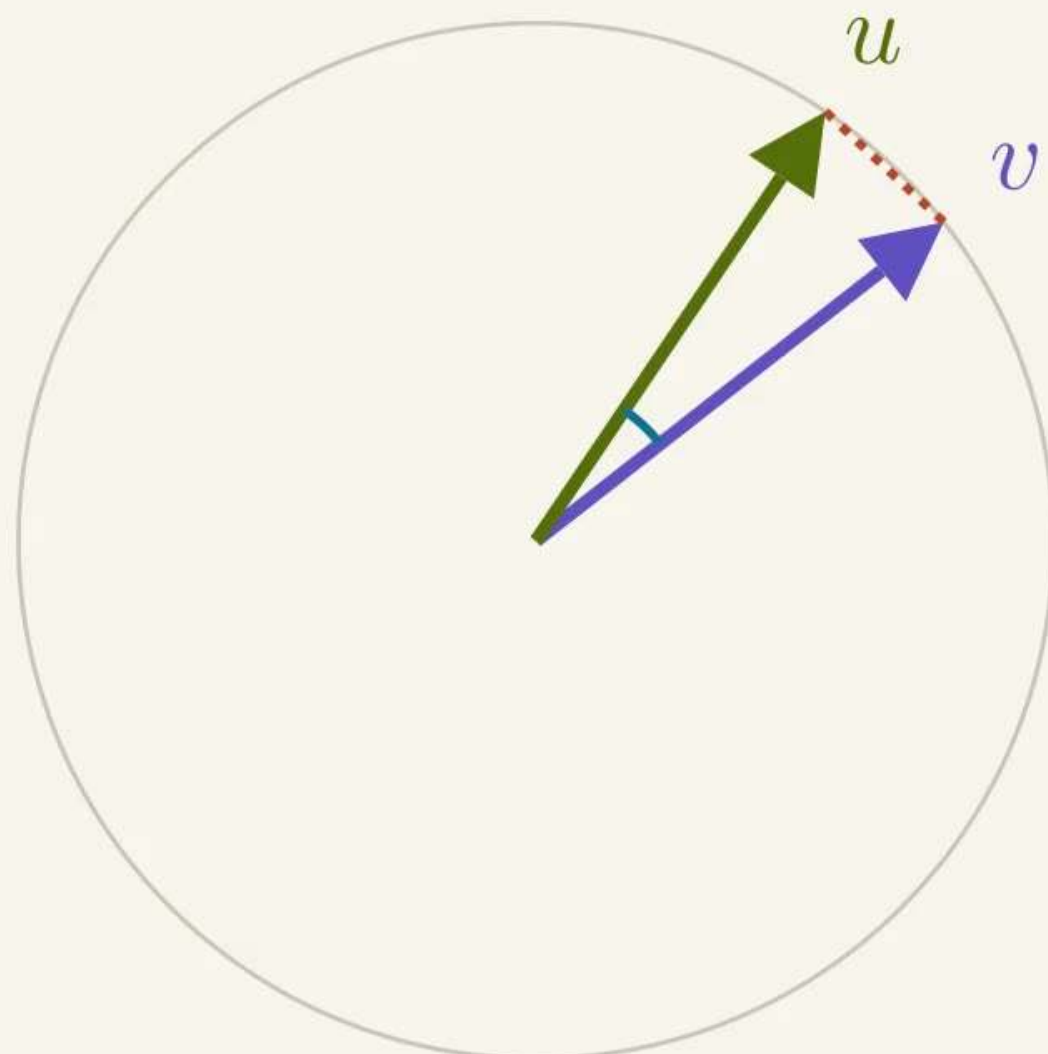
normalise, then dot product



$$s = \hat{v} \cdot \hat{u} = \cos \theta$$
$$0.95$$

one number: how aligned are the two directions?

normalise, then dot product



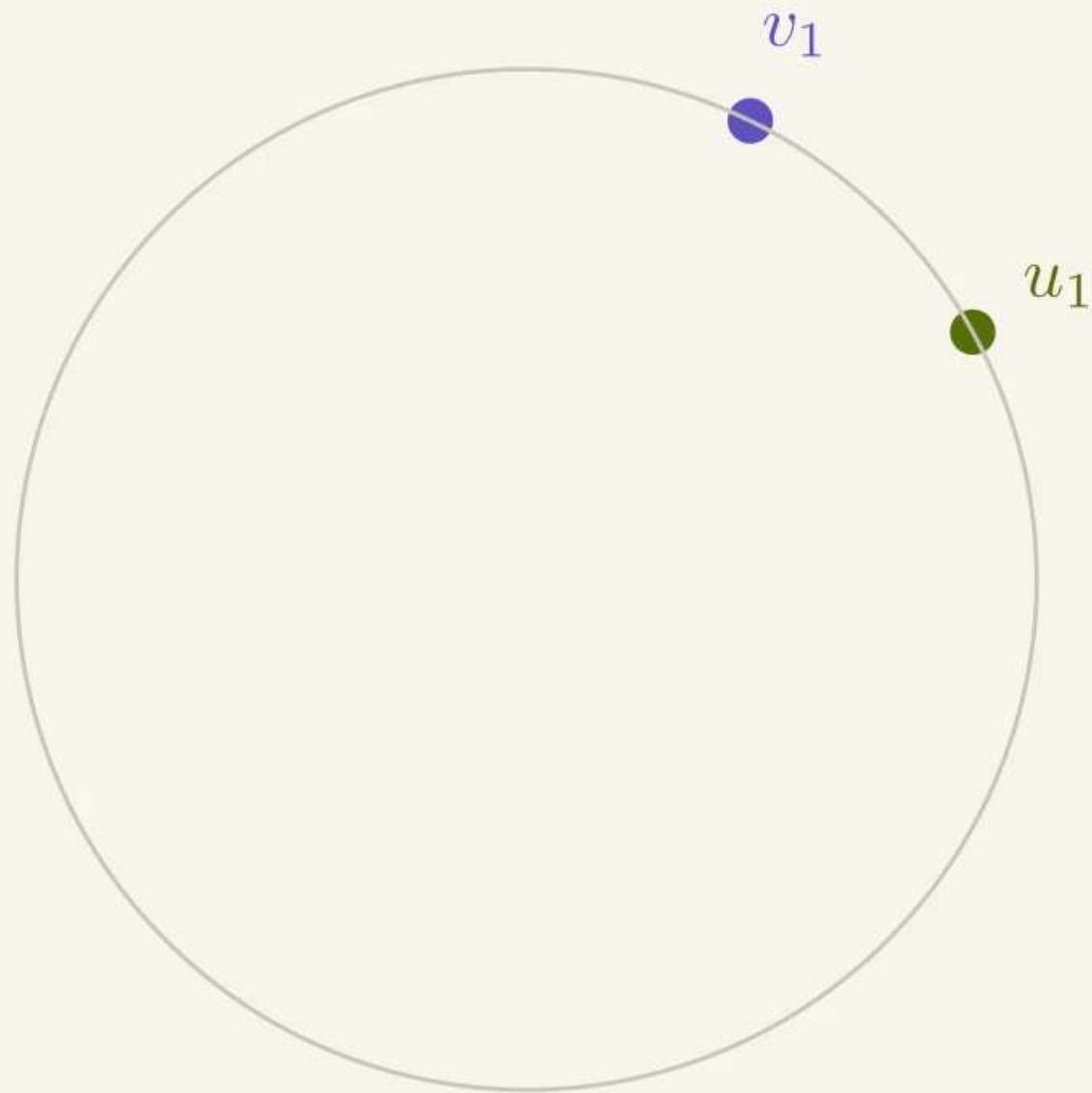
$$s = \hat{v} \cdot \hat{u} = \cos \theta$$

$$0.95$$

$$\|\hat{v} - \hat{u}\|^2 = 2 - 2 \cos \theta$$

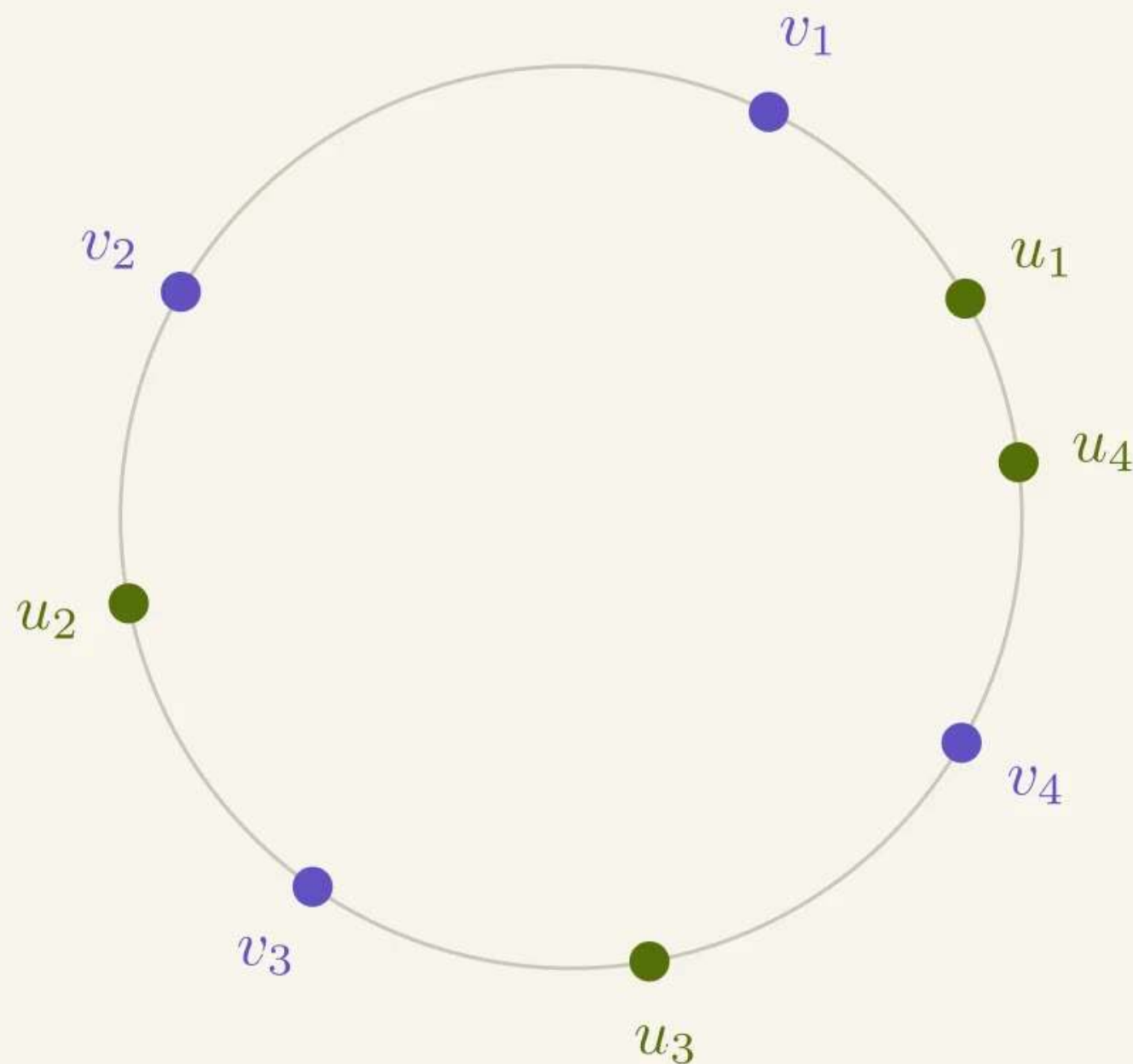
the straight-line distance ranks pairs identically — pick either

one shared space, many possible pairs



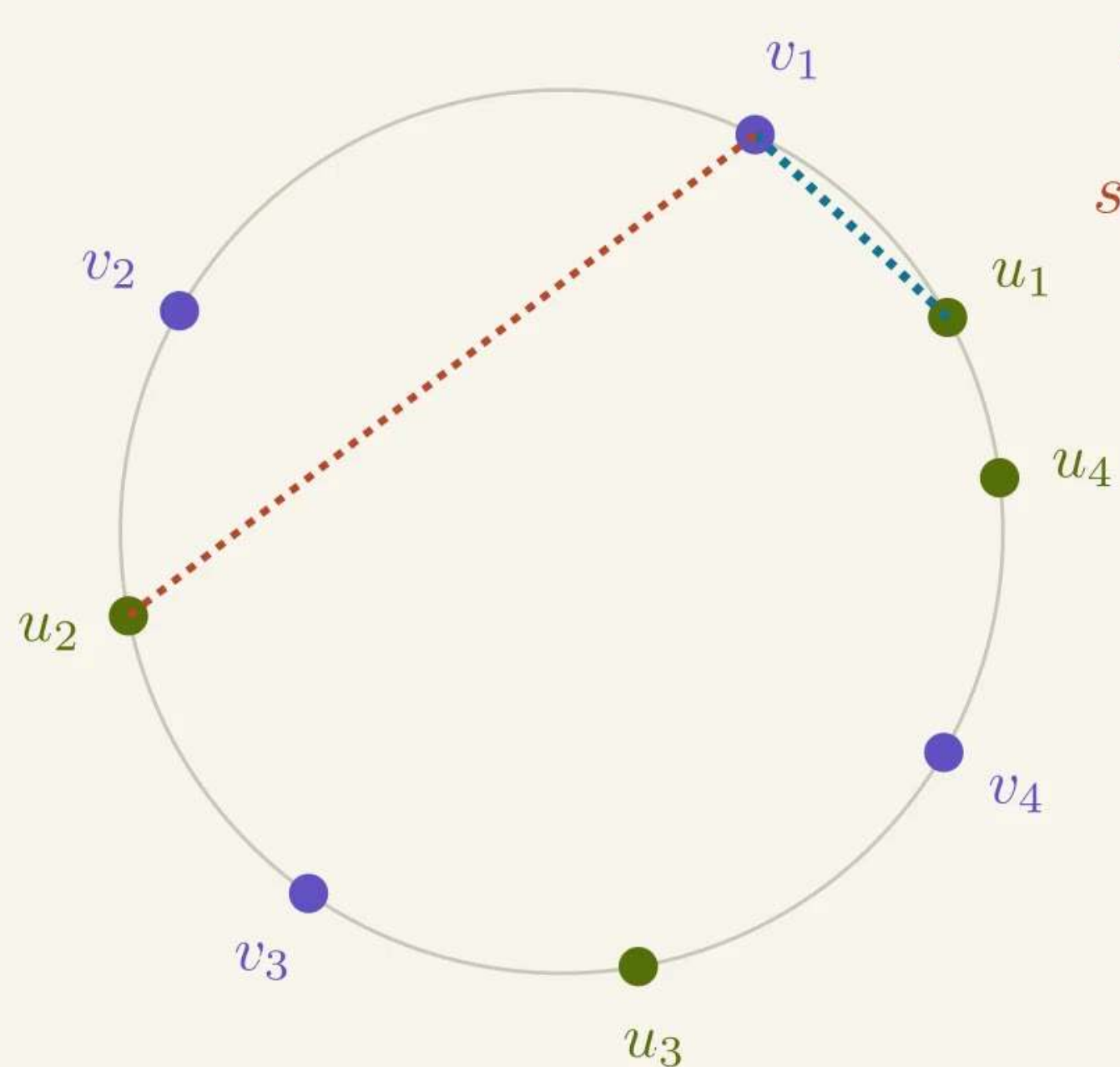
after normalisation, every embedding is a direction

one shared space, many possible pairs



four image vectors · four caption vectors · eight directions

one shared space, many possible pairs



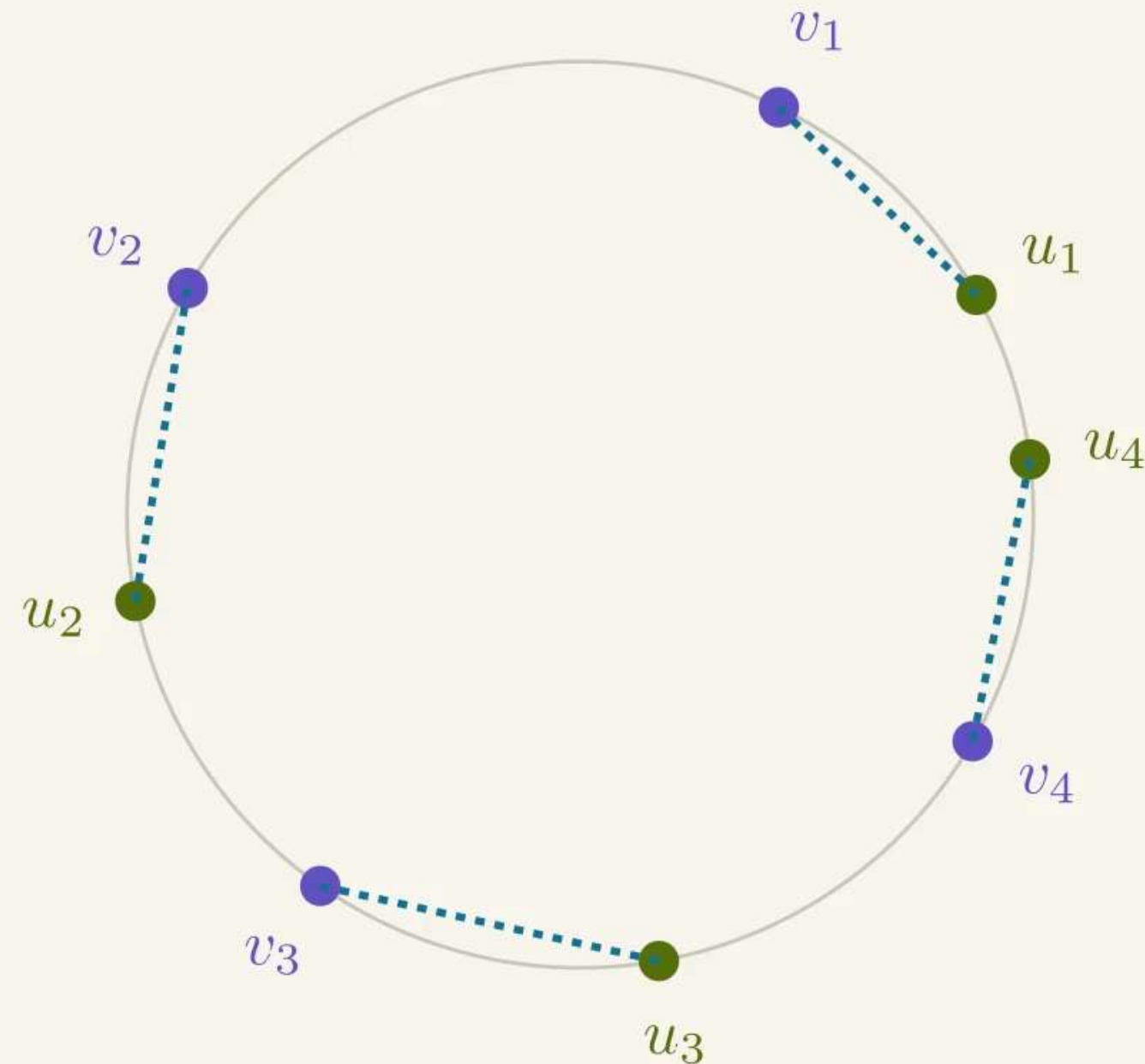
$$s_{11} = \hat{v}_1 \cdot \hat{u}_1 = 0.82$$

$$s_{12} = \hat{v}_1 \cdot \hat{u}_2 = -0.60$$

nearer directions have the larger dot product

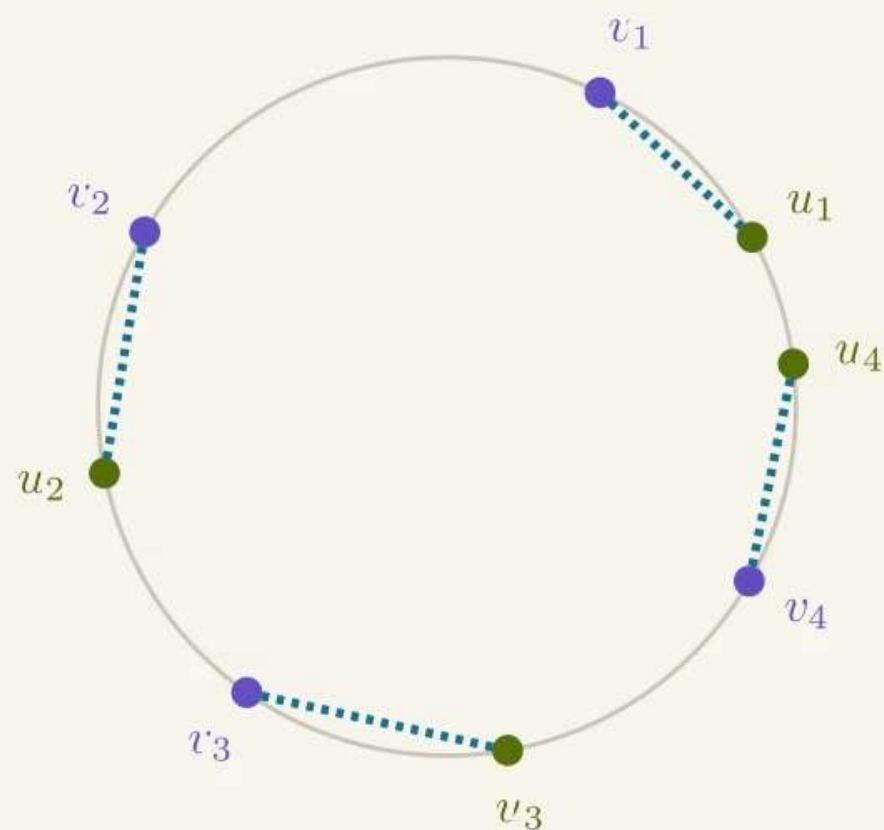
one shared space, many possible pairs

a 2D picture of a high-dimensional unit sphere



the data chooses the pairs; training pulls those directions together

vectors in; scalar scores out



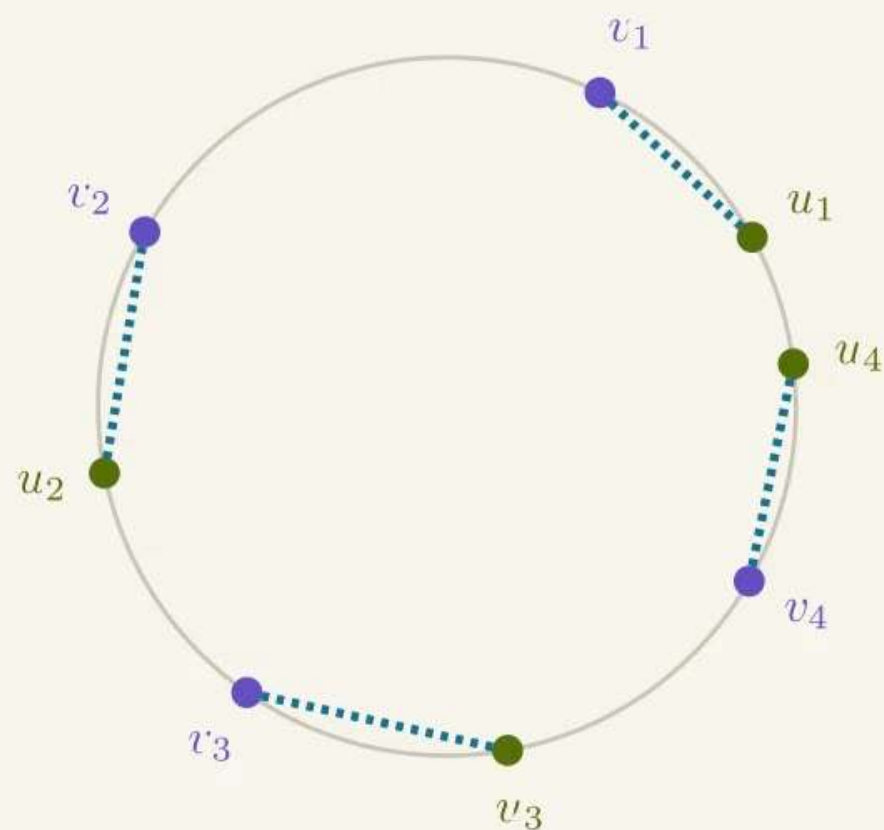
$$s_{ij} = \hat{v}_i^T \hat{u}_j$$

normalise each vector once · one pairing → one scalar

v and u are comparable because both encoders share one coordinate system

vectors in; scalar scores out

$$s_{ij} = \hat{v}_i^\top \hat{u}_j$$



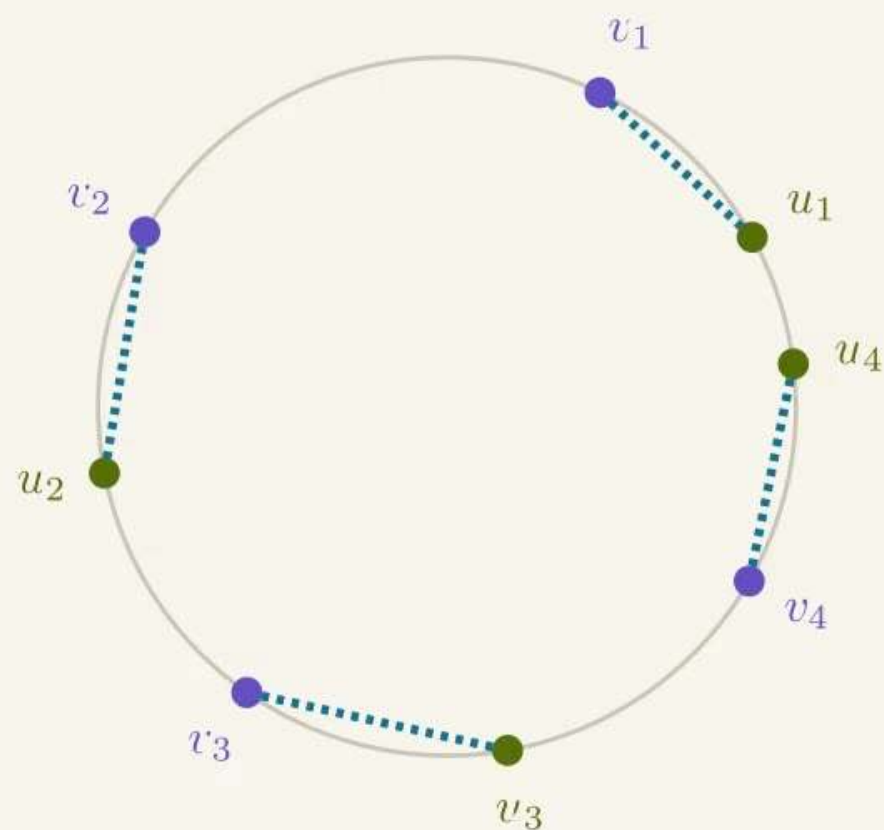
$$S = \hat{V}\hat{U}^\top = [s_{ij}]$$

	u_1	u_2	u_3	u_4
v_1	s_{11}	s_{12}	s_{13}	s_{14}
v_2	s_{21}	s_{22}	s_{23}	s_{24}
v_3	s_{31}	s_{32}	s_{33}	s_{34}
v_4	s_{41}	s_{42}	s_{43}	s_{44}

repeat the dot product for every image–caption combination

vectors in; scalar scores out

$$s_{ij} = \hat{v}_i^\top \hat{u}_j$$



$$S = \hat{V} \hat{U}^\top = [s_{ij}]$$

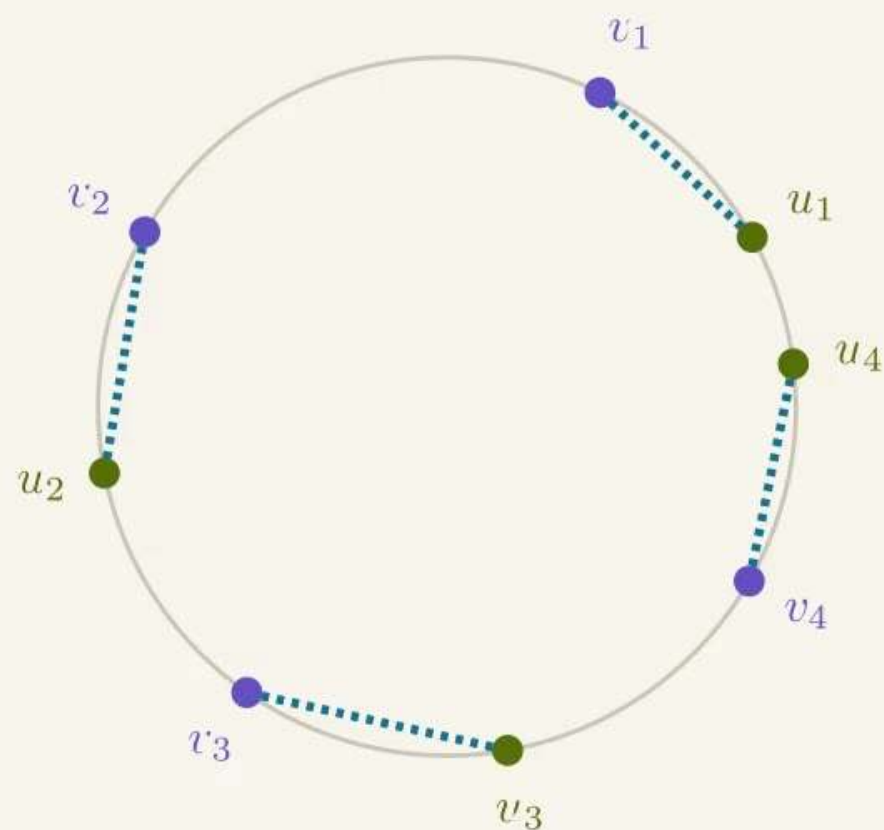
	u_1	u_2	u_3	u_4
v_1	s_{11}	s_{12}	s_{13}	s_{14}
v_2	s_{21}	s_{22}	s_{23}	s_{24}
v_3	s_{31}	s_{32}	s_{33}	s_{34}
v_4	s_{41}	s_{42}	s_{43}	s_{44}

known pairs — diagonal by indexing

the diagonal is the answer key; geometry does not put it there

vectors in; scalar scores out

$$s_{ij} = \hat{v}_i^\top \hat{u}_j$$



before training

$$S = \hat{V} \hat{U}^\top = [s_{ij}]$$

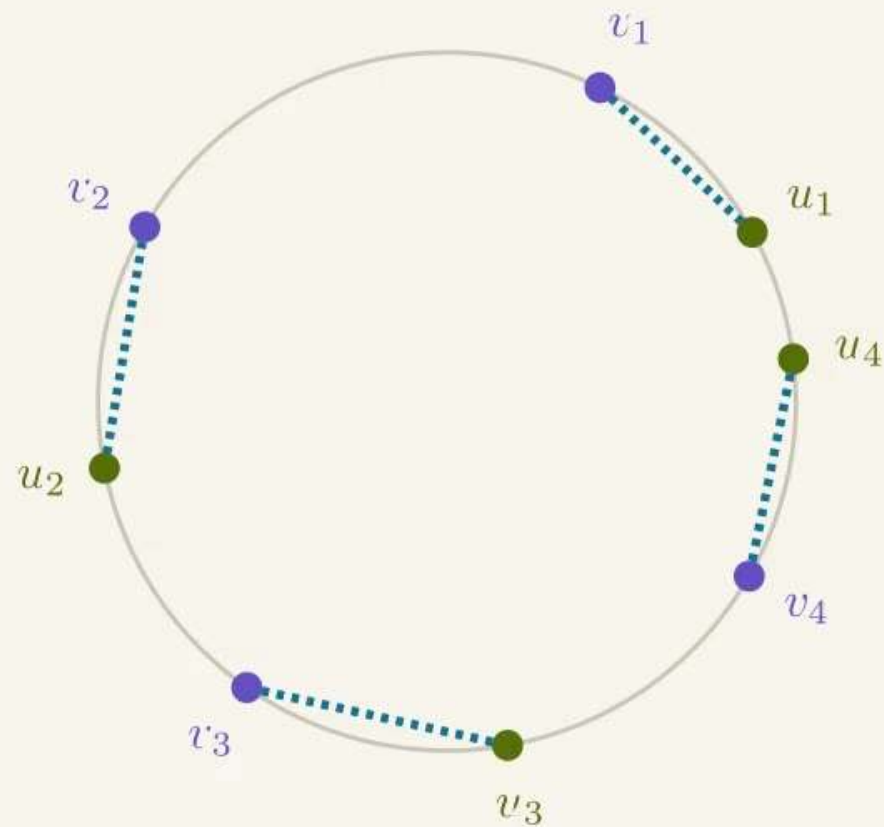
	u_1	u_2	u_3	u_4
v_1	+0.08	-0.03	+0.12	+0.01
v_2	+0.04	+0.06	-0.07	+0.11
v_3	-0.02	+0.09	+0.03	-0.05
v_4	+0.10	-0.08	+0.05	+0.00

known pairs — diagonal by indexing

the diagonal is the answer key; geometry does not put it there

vectors in; scalar scores out

$$s_{ij} = \hat{v}_i^\top \hat{u}_j$$



after training

$$S = \hat{V} \hat{U}^\top = [s_{ij}]$$

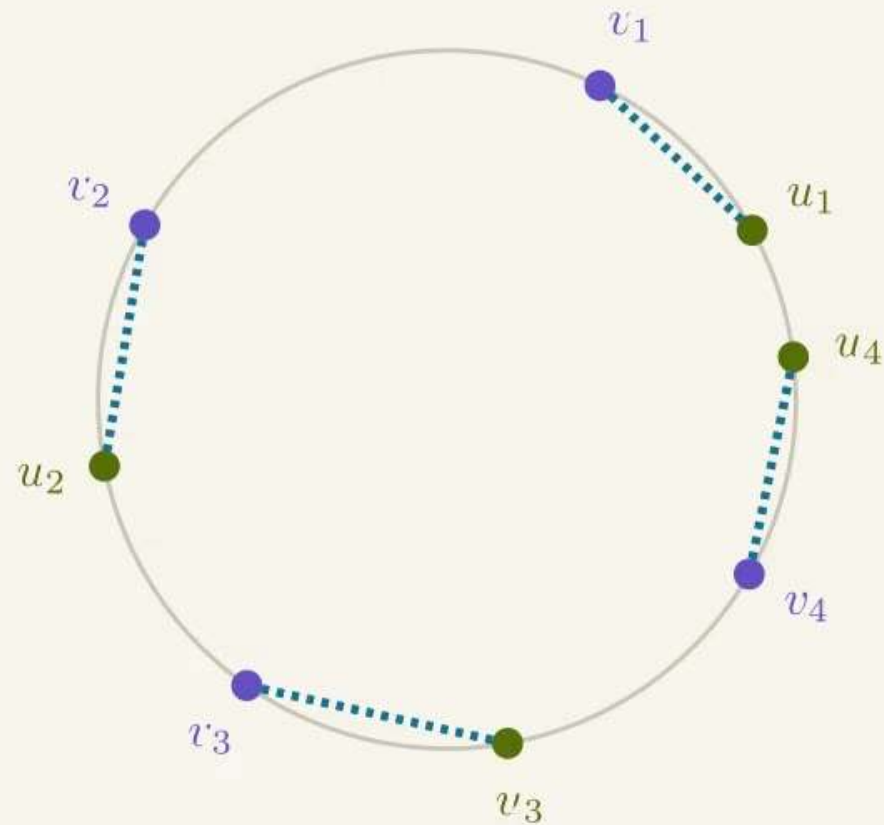
	u_1	u_2	u_3	u_4
v_1	+0.82	-0.15	+0.08	+0.03
v_2	-0.12	+0.76	+0.05	+0.11
v_3	+0.04	+0.10	+0.71	-0.09
v_4	+0.07	+0.02	-0.11	+0.80

known pairs — diagonal by indexing

training makes each target score largest in its row and column

vectors in; scalar scores out

$$s_{ij} = \hat{v}_i^\top \hat{u}_j$$



after training

$$S = \hat{V} \hat{U}^\top = [s_{ij}]$$

	u_1	u_2	u_3	u_4
v_1	+0.82	-0.15	+0.08	+0.03
v_2	-0.12	+0.76	+0.05	+0.11
v_3	+0.04	+0.10	+0.71	-0.09
v_4	+0.07	+0.02	-0.11	+0.80

$$p(j \mid I_i) = \text{softmax}_j \left(\frac{s_{ij}}{\tau} \right)$$

τ (tau) sets how sharply score gaps become probability gaps

one row is a four-way contest — the same $-\log P$ as next-word prediction

the four scores — fixed



- true pair
- near-miss
- the other captions

softmax — the contest



$$\text{loss} = -\log p(\text{true pair}) = 1.17$$

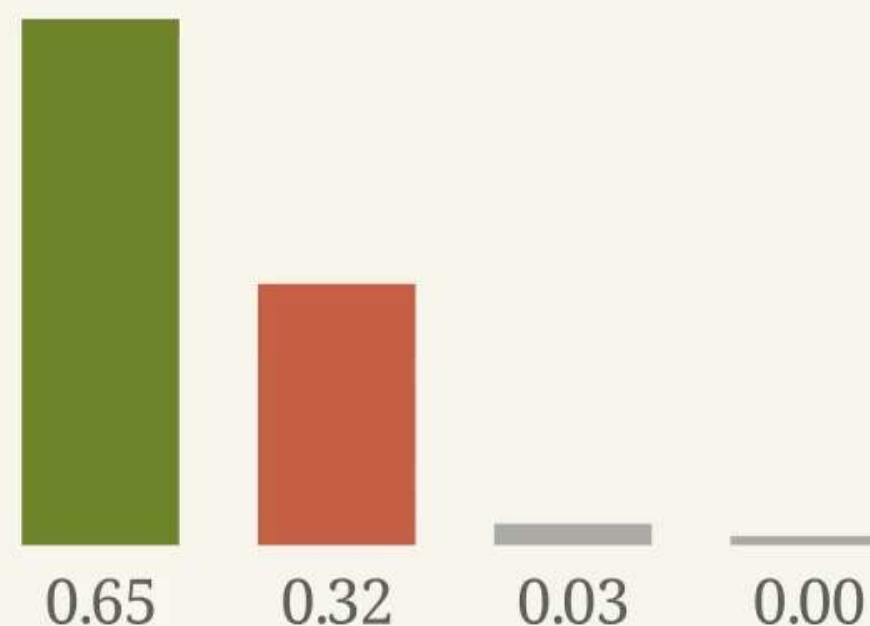
temperature τ divides the scores before the softmax

the four scores — fixed



- true pair
- near-miss
- the other captions

softmax — the contest



$$\text{loss} = -\log p(\text{true pair}) = 0.43$$

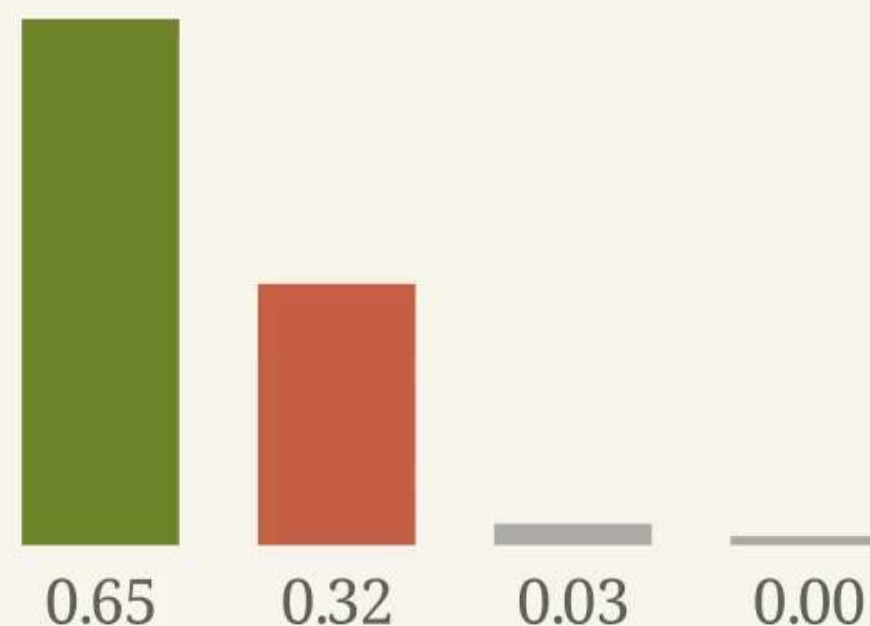
lower τ : small gaps in score become large gaps in probability

the four scores — fixed



- true pair
- near-miss
- the other captions

softmax — the contest



$$\text{loss} = -\log p(\text{true pair}) = 0.43$$

the far captions stop mattering — what is left of the loss is the near-miss

One row of the matrix, live

YOU WILL SEE

CLIP rank candidate captions for the kitchen photograph

NOTICE

swap one rival and the row changes — the image does not; then watch τ , and word order

The loss: cross-entropy over the batch

$$\mathcal{L}_i = -\log \frac{\exp(s_{ii} / \tau)}{\sum_j \exp(s_{ij} / \tau)}$$

← the true pair's score — the diagonal cell

← everyone's scores — the whole row

The loss: cross-entropy over the batch

$$\mathcal{L}_i = -\log \frac{\exp(s_{ii} / \tau)}{\sum_j \exp(s_{ij} / \tau)}$$

← the true pair's score — the diagonal cell

← everyone's scores — the whole row

—log of the probability on the true pair — the same loss as next-word prediction, with the batch as the vocabulary



I_1 — our photo

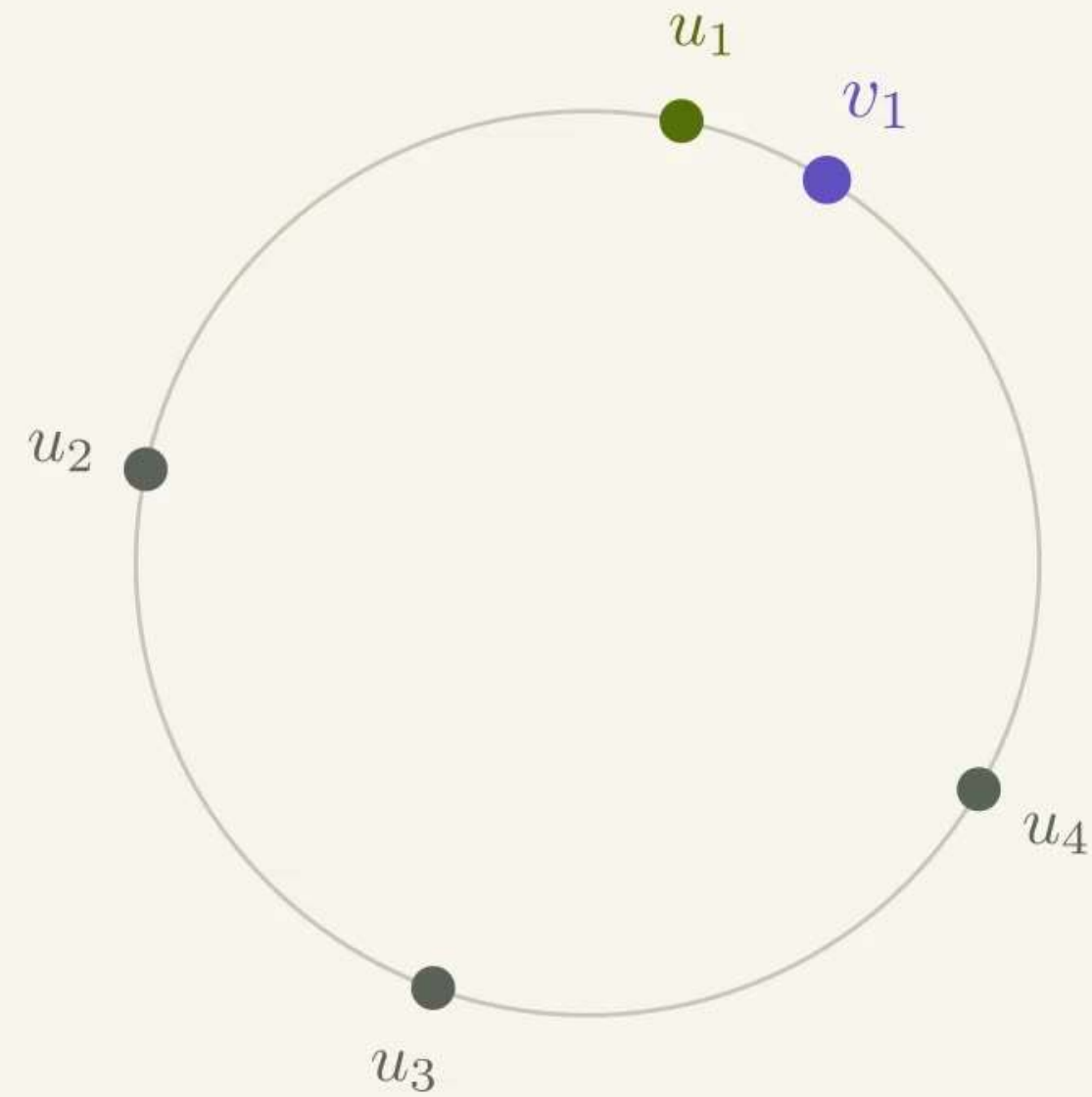
"a red cup beside a blue cup"
arrived with I_1 — the positive

"a dog on the beach"
someone else's caption — a negative

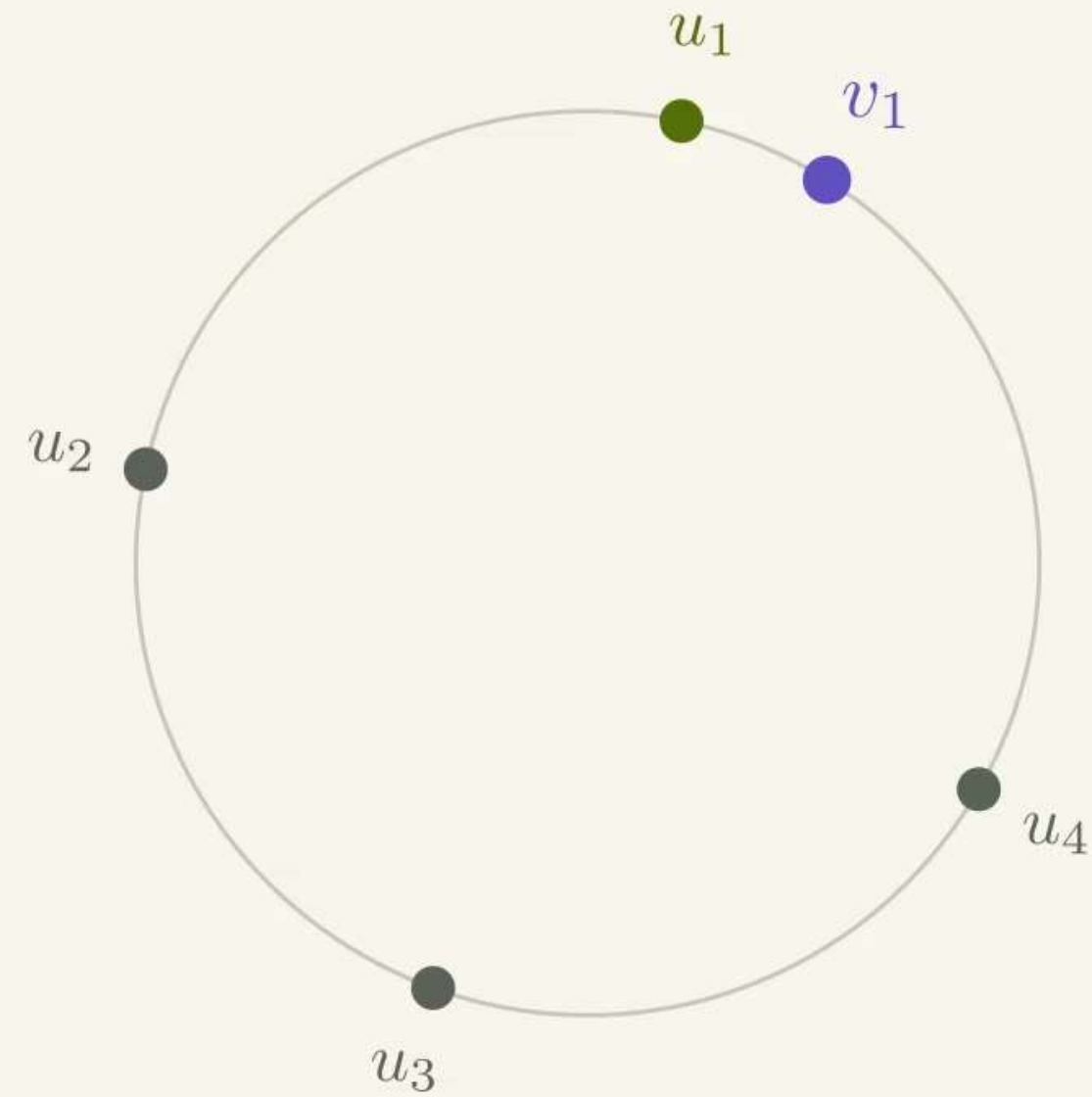
"a city bus at night"
a negative

"a violinist on stage"
a negative

a negative is not a property — it is a pairing the batch never made



every training step: pull the positive in — push every negative away



the push is the anti-collapse force — remove the negatives and nothing resists

The positive pair decides what counts as the same



one photograph

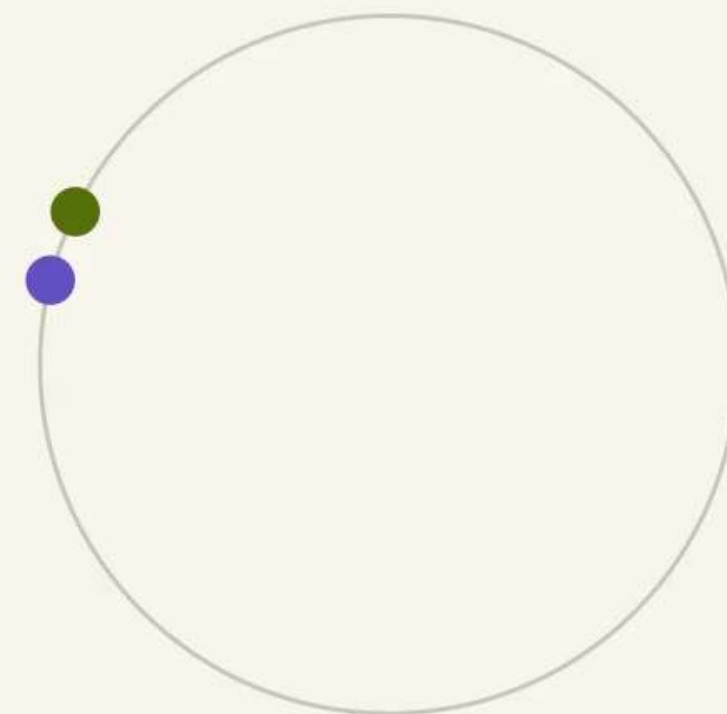


crop

=



colour jitter · rotate · blur



their two embeddings

two altered views of one photo — the loss says: the same

The positive pair decides what counts as the same



one photograph

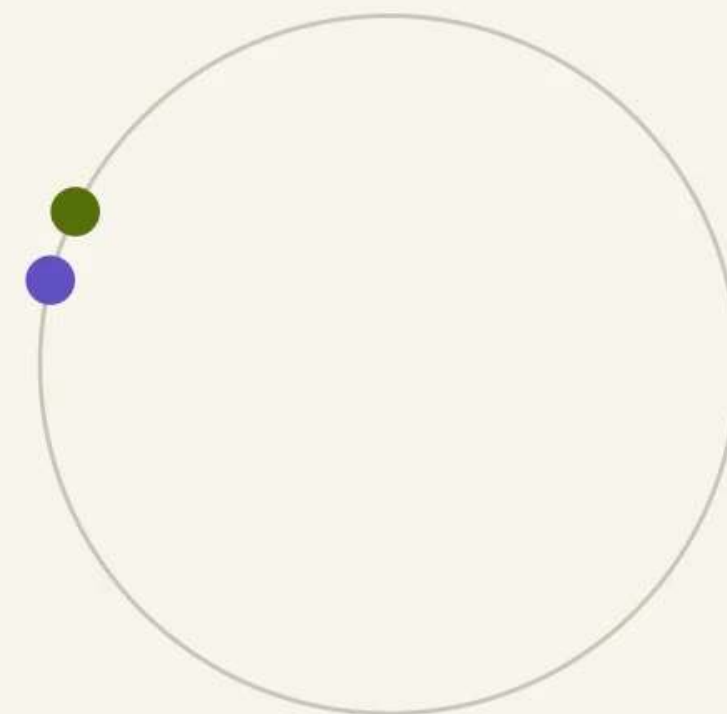


crop

=



colour jitter · rotate · blur



their two embeddings

preserved: object identity — still a red cup and a blue cup in a sink

The positive pair decides what counts as the same



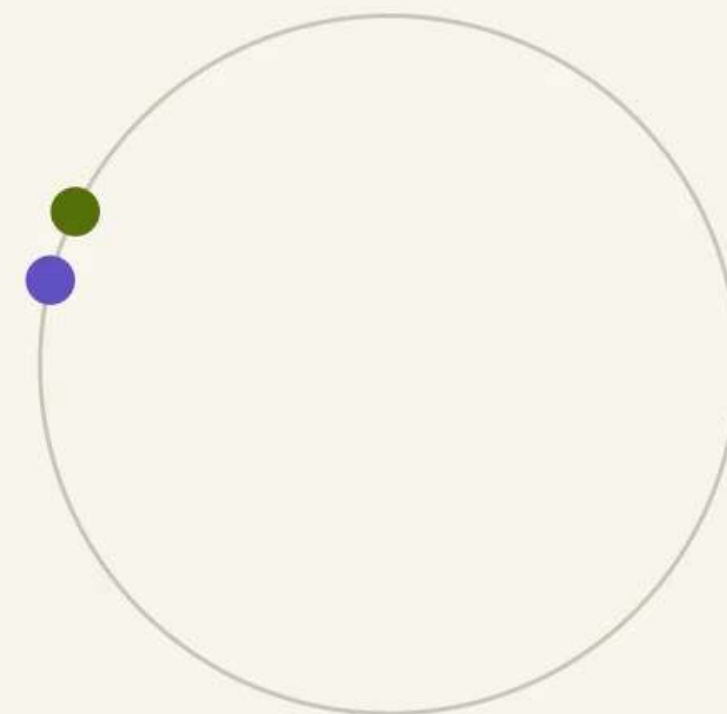
one photograph



crop
=
=



colour jitter · rotate · blur



their two embeddings

ignored: colour, framing, and what the crop cut away — the gripper

The positive pair decides what counts as the same



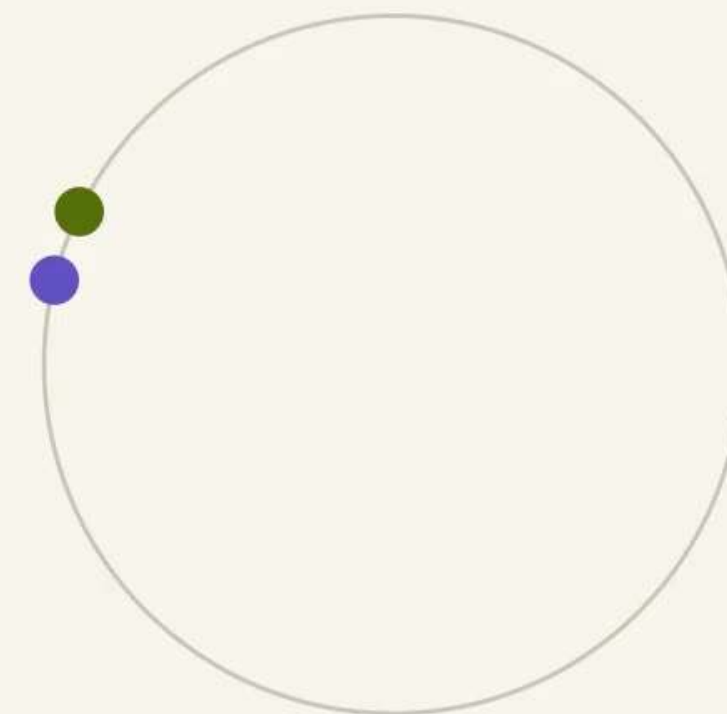
one photograph



crop
=
=



colour jitter · rotate · blur



their two embeddings

conflated: this red cup with a greener one; this framing with any other

The positive pair decides what counts as the same

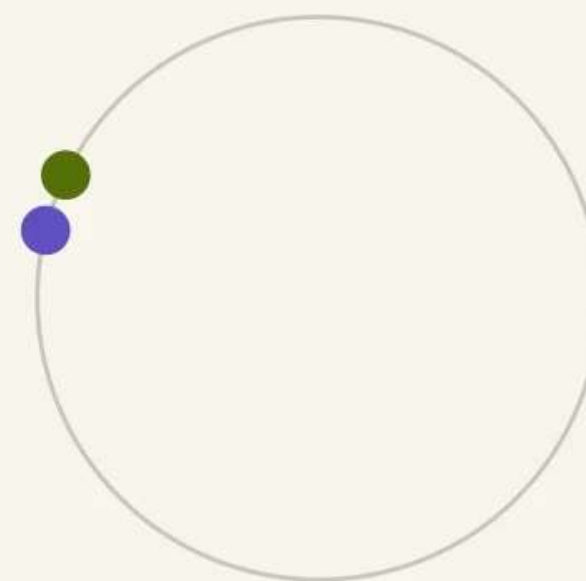


the photograph

=

a red cup beside a blue cup in the sink

what the caption names



their two embeddings

image and caption — the loss says: they describe the same event

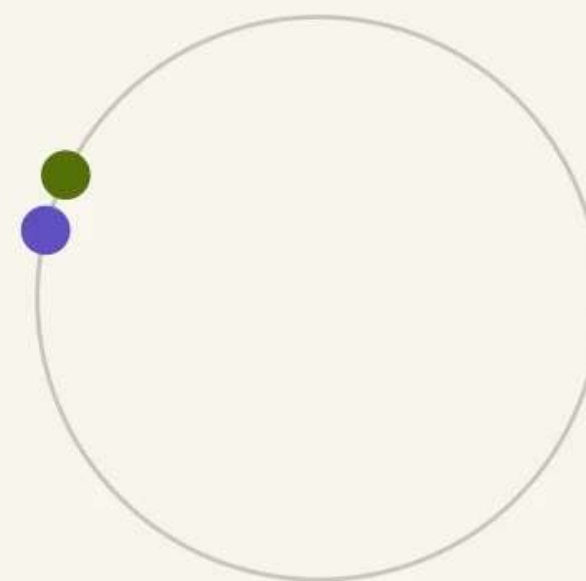
The positive pair decides what counts as the same



the photograph

= a red cup beside a blue cup in the sink

what the caption names



their two embeddings

preserved: named objects and relations — red cup, blue cup, beside, sink

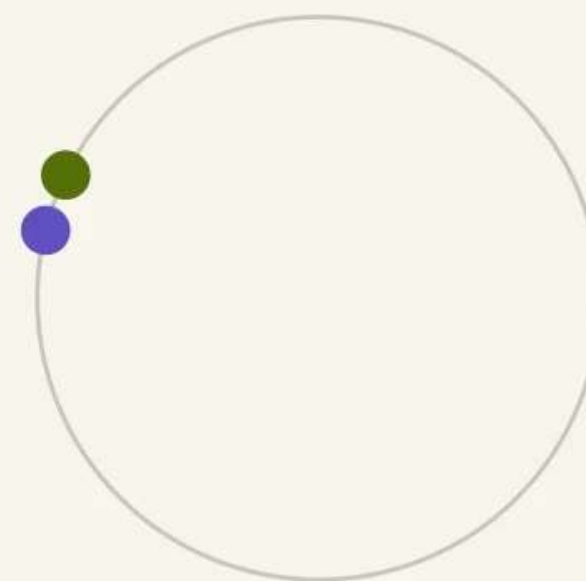
The positive pair decides what counts as the same



the photograph

= a red cup beside a blue cup in the sink

what the caption names



their two embeddings

ignored: what the caption never says — handle angle, depth, wet or dry

The positive pair decides what counts as the same

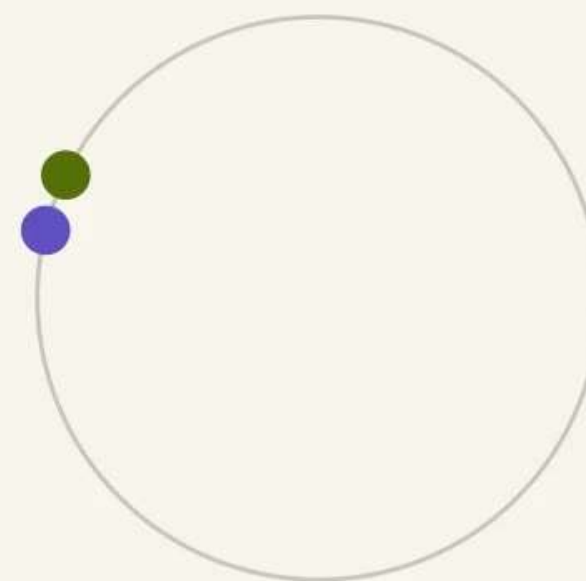


the photograph

=

a red cup beside a blue cup in the sink

what the caption names



their two embeddings

conflated: every scene those seven words fit

Three kinds of negative

ANCHOR



"a red cup beside a blue cup"

Three kinds of negative

ANCHOR



"a red cup beside a blue cup"

USEFUL RIVAL

"a red bowl beside a blue cup"

forces object discrimination

Three kinds of negative

ANCHOR



"a red cup beside a blue cup"

USEFUL RIVAL

"a red bowl beside a blue cup"

forces object discrimination

FALSE NEGATIVE

another valid caption for this scene

pushes away a correct answer

Three kinds of negative

ANCHOR



"a red cup beside a blue cup"

USEFUL RIVAL

"a red bowl beside a blue cup"

forces object discrimination

FALSE NEGATIVE

another valid caption for this scene

pushes away a correct answer

EASY NEGATIVE

"a satellite over the Pacific"

already separated; almost no gradient

Three kinds of negative

ANCHOR



"a red cup beside a blue cup"

USEFUL RIVAL

"a red bowl beside a blue cup"

forces object discrimination

FALSE NEGATIVE

another valid caption for this scene

pushes away a correct answer

EASY NEGATIVE

"a satellite over the Pacific"

already separated; almost no gradient

The comparison set decides which differences receive pressure.

Contrastive and regularised joint embedding



Contrastive and regularised joint embedding



CONTRASTIVE

match the pair + repel alternatives

Contrastive and regularised joint embedding



CONTRASTIVE

match the pair + repel alternatives

REGULARISED / PREDICTIVE

match or predict the target — no rivals

Contrastive and regularised joint embedding



CONTRASTIVE

match the pair + repel alternatives

COMPARES AGAINST every rival in the batch

REGULARISED / PREDICTIVE

match or predict the target — no rivals

Contrastive and regularised joint embedding



CONTRASTIVE

match the pair + repel alternatives

COMPARES AGAINST every rival in the batch

REGULARISED / PREDICTIVE

match or predict the target — no rivals

COMPARES AGAINST the pair alone, or a predicted target

Contrastive and regularised joint embedding



CONTRASTIVE

match the pair + repel alternatives

COMPARES AGAINST every rival in the batch

COLLAPSE STOPPED BY the rivals themselves — repulsion

REGULARISED / PREDICTIVE

match or predict the target — no rivals

COMPARES AGAINST the pair alone, or a predicted target

Contrastive and regularised joint embedding



CONTRASTIVE

match the pair + repel alternatives

COMPARES AGAINST every rival in the batch

COLLAPSE STOPPED BY the rivals themselves — repulsion

REGULARISED / PREDICTIVE

match or predict the target — no rivals

COMPARES AGAINST the pair alone, or a predicted target

COLLAPSE STOPPED BY asymmetry · variance · decorrelation

Contrastive and regularised joint embedding



CONTRASTIVE

match the pair + repel alternatives

COMPARES AGAINST every rival in the batch

COLLAPSE STOPPED BY the rivals themselves — repulsion

LOSS softmax over similarities · InfoNCE

REGULARISED / PREDICTIVE

match or predict the target — no rivals

COMPARES AGAINST the pair alone, or a predicted target

COLLAPSE STOPPED BY asymmetry · variance · decorrelation

Contrastive and regularised joint embedding



CONTRASTIVE

match the pair + repel alternatives

COMPARES AGAINST every rival in the batch

COLLAPSE STOPPED BY the rivals themselves — repulsion

LOSS softmax over similarities · InfoNCE

REGULARISED / PREDICTIVE

match or predict the target — no rivals

COMPARES AGAINST the pair alone, or a predicted target

COLLAPSE STOPPED BY asymmetry · variance · decorrelation

LOSS regression distance (L1/L2) + regularisers

Contrastive and regularised joint embedding



CONTRASTIVE

match the pair + repel alternatives

COMPARES AGAINST every rival in the batch

COLLAPSE STOPPED BY the rivals themselves — repulsion

LOSS softmax over similarities · InfoNCE

PRICE huge batches · false negatives · batch-shaped geometry

REGULARISED / PREDICTIVE

match or predict the target — no rivals

COMPARES AGAINST the pair alone, or a predicted target

COLLAPSE STOPPED BY asymmetry · variance · decorrelation

LOSS regression distance (L1/L2) + regularisers

Contrastive and regularised joint embedding



CONTRASTIVE

match the pair + repel alternatives

COMPARES AGAINST every rival in the batch

COLLAPSE STOPPED BY the rivals themselves — repulsion

LOSS softmax over similarities · InfoNCE

PRICE huge batches · false negatives · batch-shaped geometry

REGULARISED / PREDICTIVE

match or predict the target — no rivals

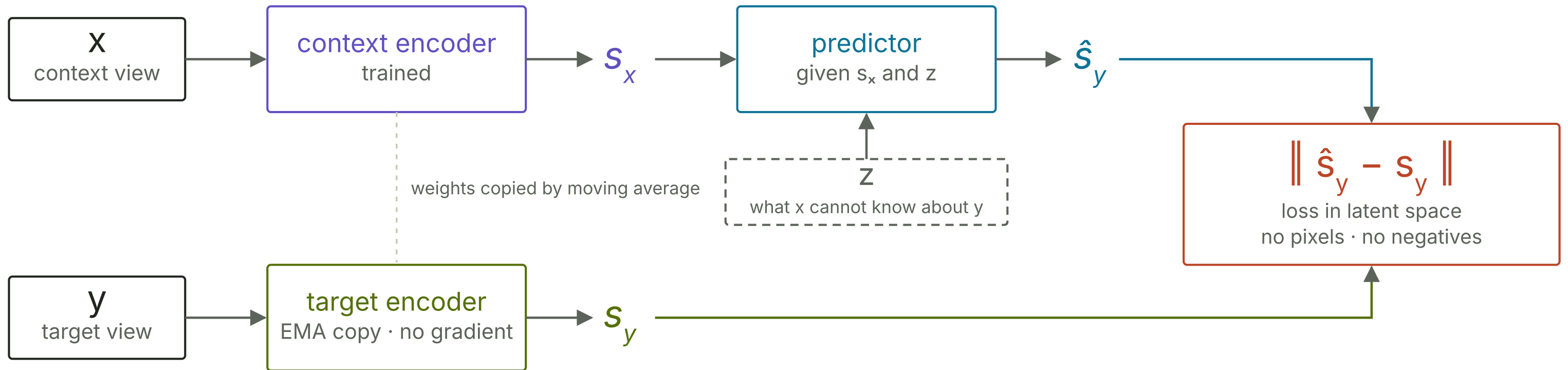
COMPARES AGAINST the pair alone, or a predicted target

COLLAPSE STOPPED BY asymmetry · variance · decorrelation

LOSS regression distance (L1/L2) + regularisers

PRICE assumptions about the representation's shape

JEPA: predict the target's representation



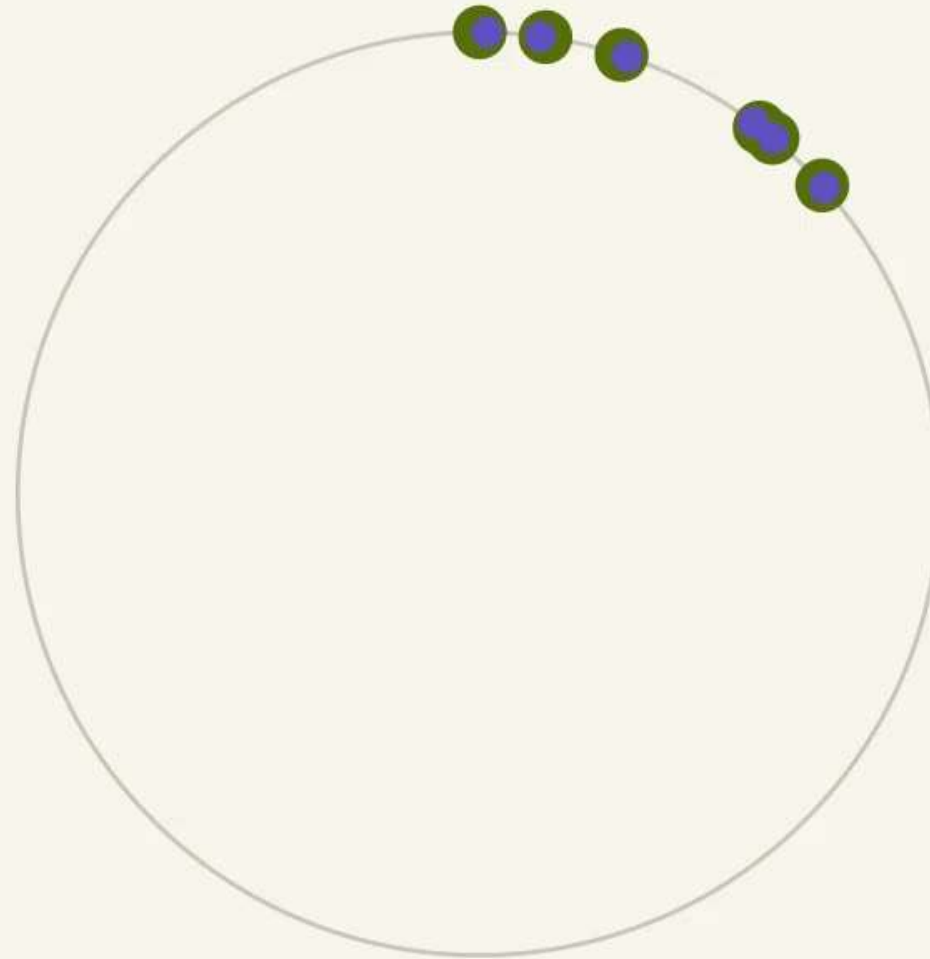
collapse is prevented by the asymmetry (stop-gradient / EMA target) or by a term on the batch of embeddings (variance + covariance, VICReg; a distribution test, SIGReg) — not by rivals

*No pixels are reconstructed and no rivals are compared: the predictor is asked for the target's **representation**, and the target encoder is a slowly-moving copy that does not chase back.*

a lazy start: everything in one clump — the objective only pulls pairs together

● image
● caption

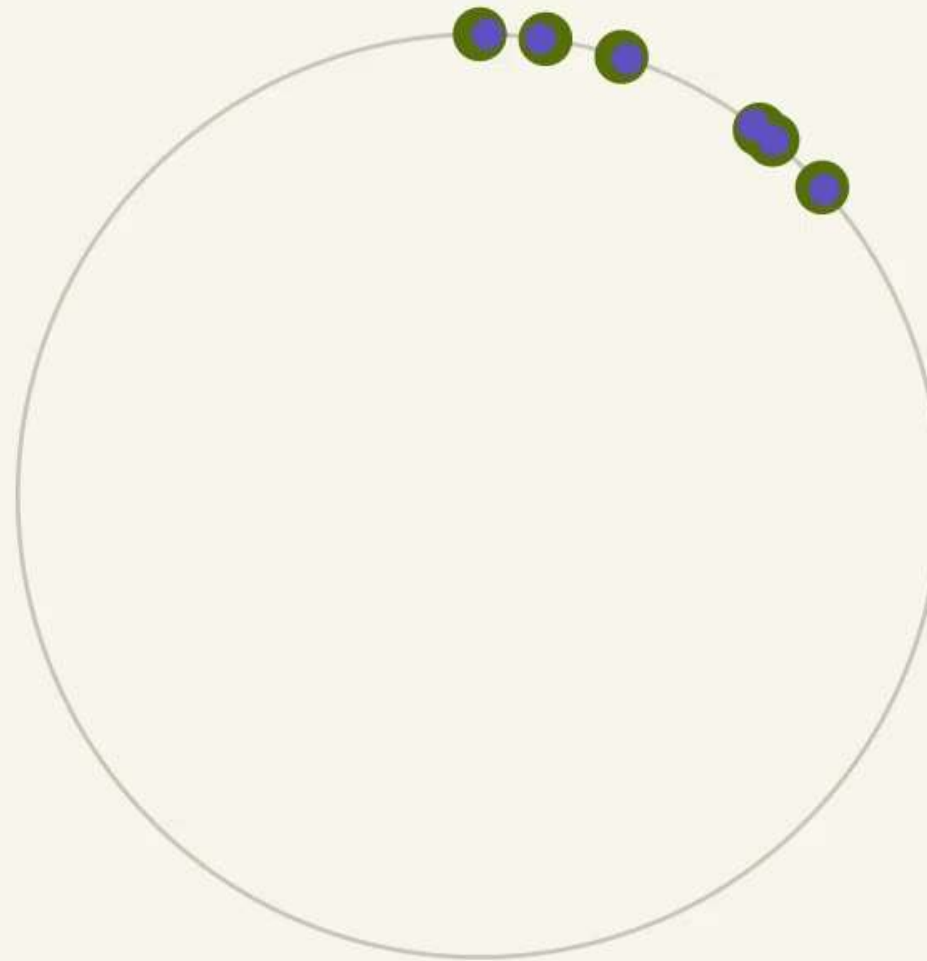
training step 50



a lazy start: everything in one clump — the objective only pulls pairs together

● image
● caption

training step 50

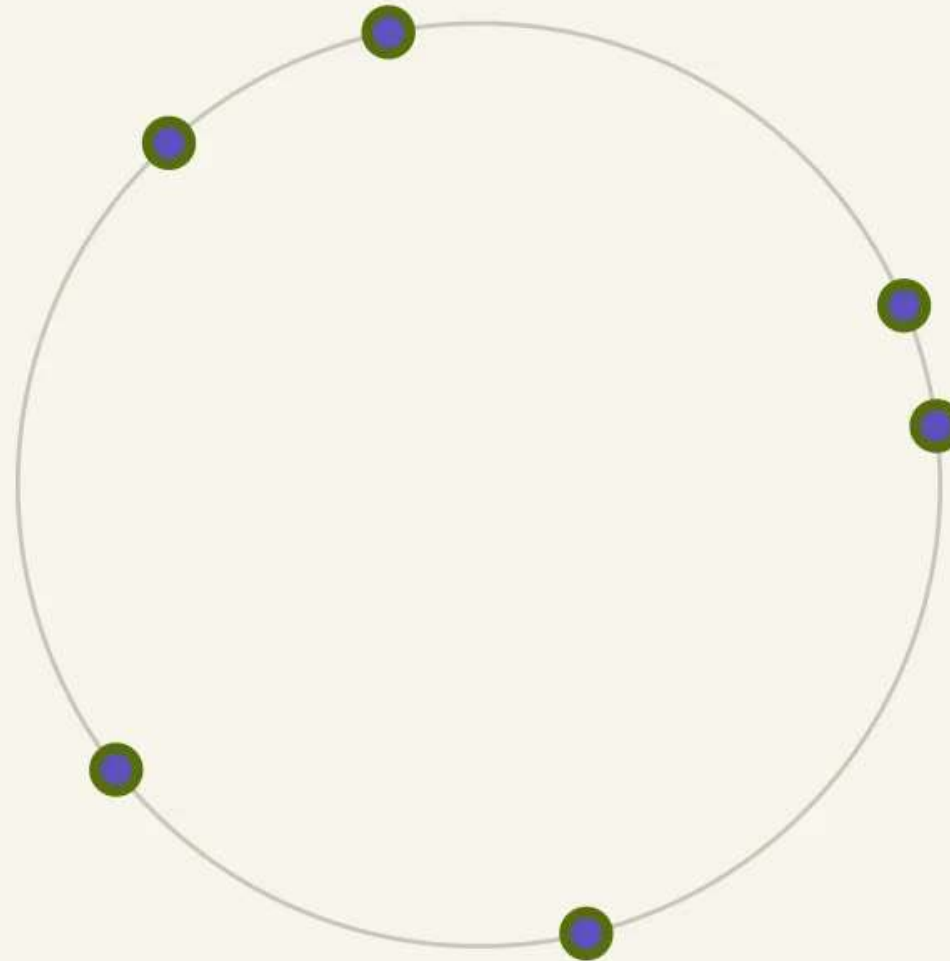


pairs meet — and the clump stays a clump: agreement is satisfied, information is not

same clumped start — no negatives: agreement plus a variance term on the batch

- image
- caption

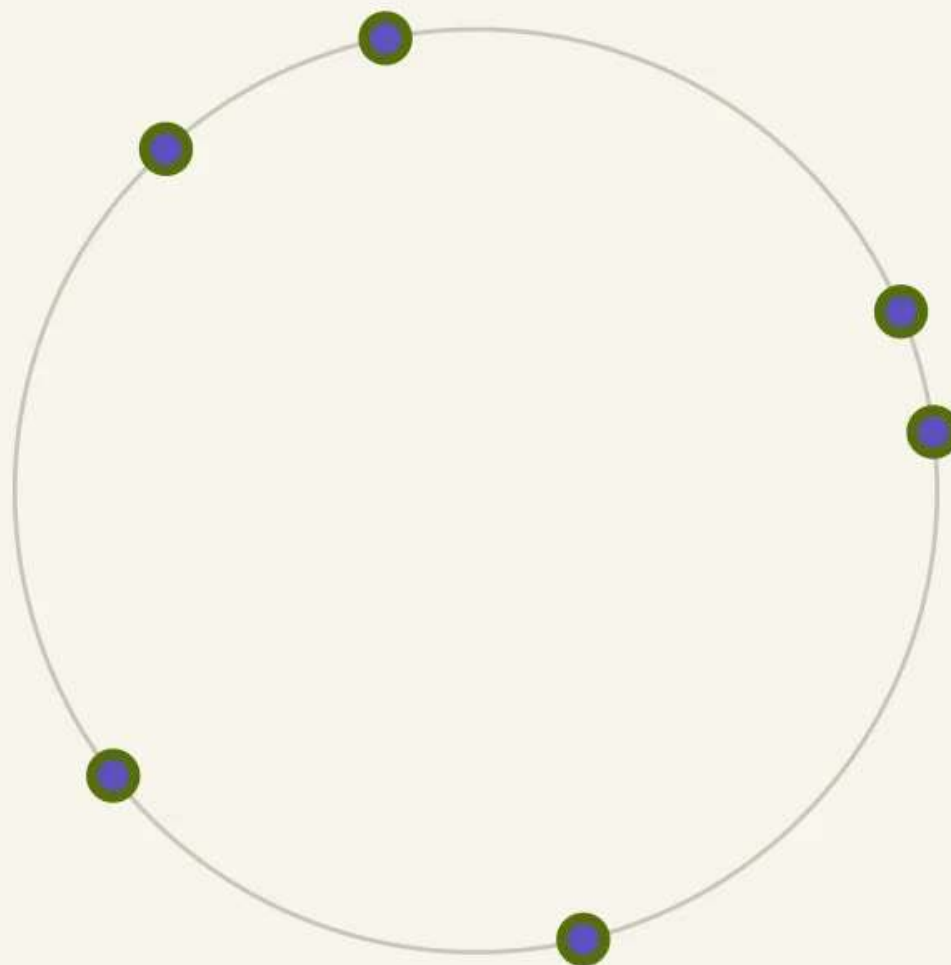
training step 140



same clumped start — no negatives: agreement plus a variance term on the batch

● image
● caption

training step 140



flat dimensions are punished: the space opens without a single comparison — VICReg · SIGReg

Preventing collapse without negatives

THE FAILURE TO PREVENT

everything → same
vector

perfect agreement
zero information

Preventing collapse without negatives

THE FAILURE TO PREVENT

everything → same
vector

perfect agreement
zero information

ASYMMETRY

predictor → stop-gradient teacher

one side doesn't chase the other

Preventing collapse without negatives

THE FAILURE TO PREVENT

everything → same
vector

perfect agreement
zero information

ASYMMETRY

predictor → stop-gradient teacher

one side doesn't chase the other

VARIANCE

keep dimensions spread out

penalise any dimension that goes constant

Preventing collapse without negatives

THE FAILURE TO PREVENT

everything → same
vector

perfect agreement
zero information

ASYMMETRY

predictor → stop-gradient teacher

one side doesn't chase the other

VARIANCE

keep dimensions spread out

penalise any dimension that goes constant

DECORRELATION

stop dimensions copying each other

covariance regularisation

The embedding keeps what the objective needs

CONTRASTIVE

keeps what tells the pairs apart

associations that help identify the correct pair get pressure ·
everything else is incidental — protected only if it happens to help

The embedding keeps what the objective needs

CONTRASTIVE

keeps what tells the pairs apart

associations that help identify the correct pair get pressure ·
everything else is incidental — protected only if it happens to help

PREDICTIVE JOINT EMBEDDING

keeps what predicts the target

the prediction objective is a signal that pulls useful information back
into the embedding · what it never has to predict, it can drop

The embedding keeps what the objective needs

CONTRASTIVE

keeps what tells the pairs apart

associations that help identify the correct pair get pressure ·
everything else is incidental — protected only if it happens to help

PREDICTIVE JOINT EMBEDDING

keeps what predicts the target

the prediction objective is a signal that pulls useful information back
into the embedding · what it never has to predict, it can drop

Crop · colour jitter · blur declared colour and framing incidental. Then the instruction arrives: "pick up the red mug by its handle."

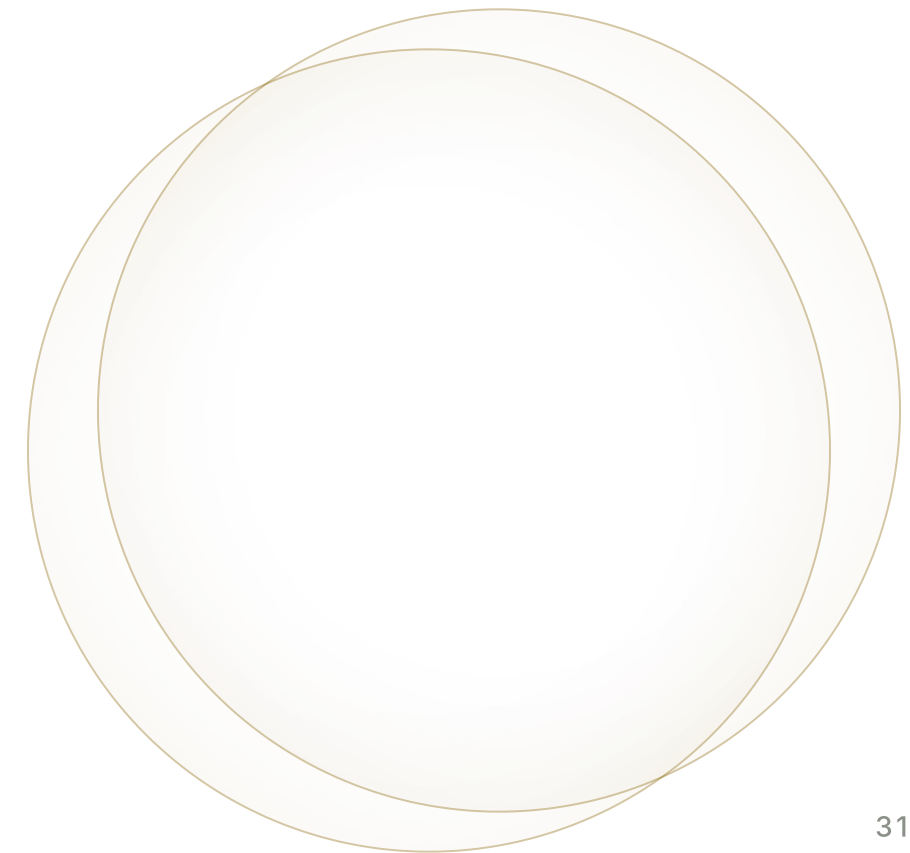
What does DINOv2 keep?

YOU WILL SEE

3,072 patch vectors over the kitchen image

NOTICE

which other patches DINOv2 encoded like
the one you select



What sits on top of the encoder

THE BASE

a self-supervised encoder

contrastive or regularised joint embedding — no labels, no rewards;
it keeps what its objective needed

What sits on top of the encoder

THE BASE

a self-supervised encoder

contrastive or regularised joint embedding — no labels, no rewards;
it keeps what its objective needed

WHERE REWARDS CAN BE CHECKED

reinforcement learning from verifiable rewards

on top of that encoder: code that compiles, a proof that checks, a
game that is won, a simulator that scores — millions of cheap trials

What sits on top of the encoder

THE BASE

a self-supervised encoder

contrastive or regularised joint embedding — no labels, no rewards; it keeps what its objective needed

IN A MESSY PHYSICAL DOMAIN

RLVR is expensive

every trial costs real time, hardware and a reset; "did it succeed?" is often not verifiable without a person; exploration breaks things

WHERE REWARDS CAN BE CHECKED

reinforcement learning from verifiable rewards

on top of that encoder: code that compiles, a proof that checks, a game that is won, a simulator that scores — millions of cheap trials

What sits on top of the encoder

THE BASE

a self-supervised encoder

contrastive or regularised joint embedding — no labels, no rewards; it keeps what its objective needed

IN A MESSY PHYSICAL DOMAIN

RLVR is expensive

every trial costs real time, hardware and a reset; "did it succeed?" is often not verifiable without a person; exploration breaks things

So VLA models mostly learn from human data: demonstrations, teacher-forced — predict the action the person took, the same self-supervised move as the next word.

WHERE REWARDS CAN BE CHECKED

reinforcement learning from verifiable rewards

on top of that encoder: code that compiles, a proof that checks, a game that is won, a simulator that scores — millions of cheap trials

Four questions for any representation objective

pair

What counts as belonging together?

compare

Against what alternatives must the match win?

prevent collapse

What keeps the representation from going constant?

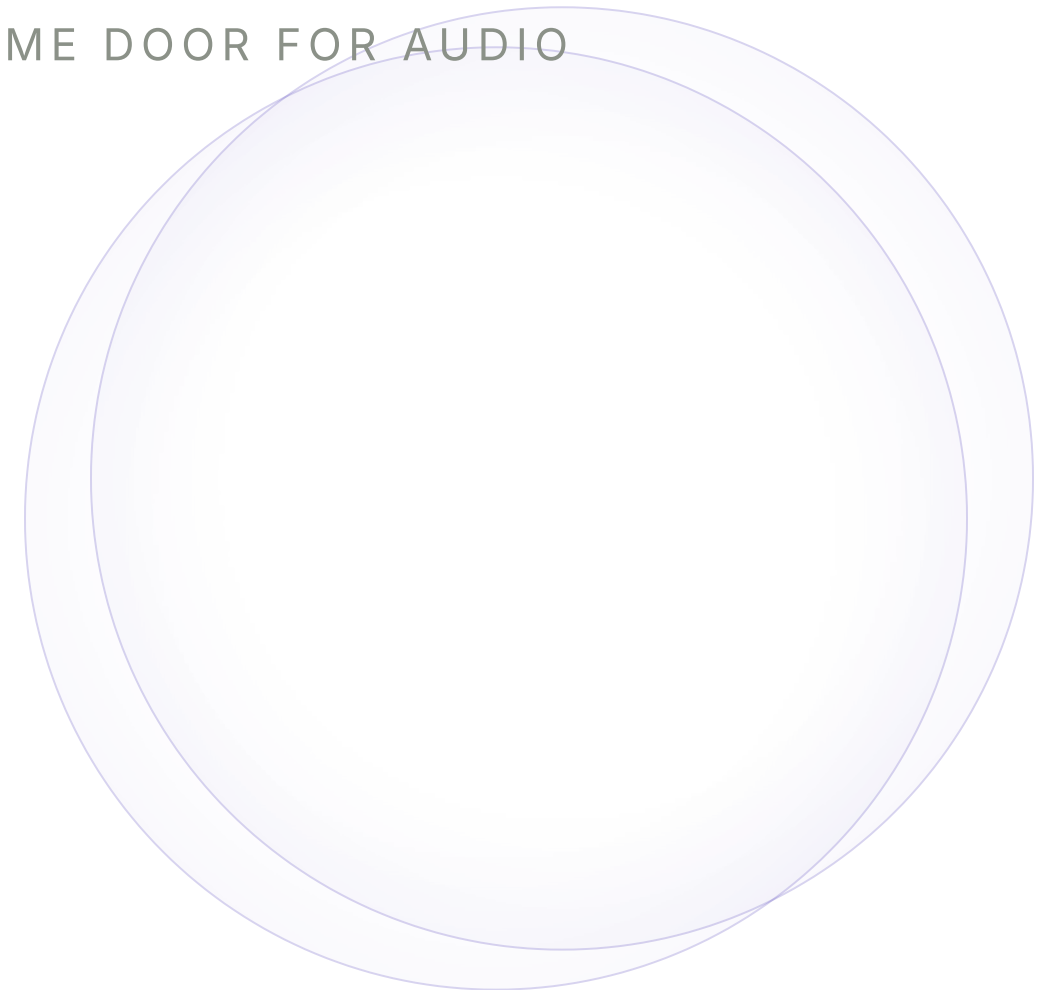
retain

What must survive for the downstream task?

Signals into the language model

Vision, then audio, as input to a language model

PATCHES → VISUAL TOKENS → ONE SEQUENCE → AN ANSWER · THEN THE SAME DOOR FOR AUDIO



The language model's interface is a sequence of vectors

WHAT YOU TYPE

"Which cup did she mean?"



The language model's interface is a sequence of vectors

WHAT YOU TYPE

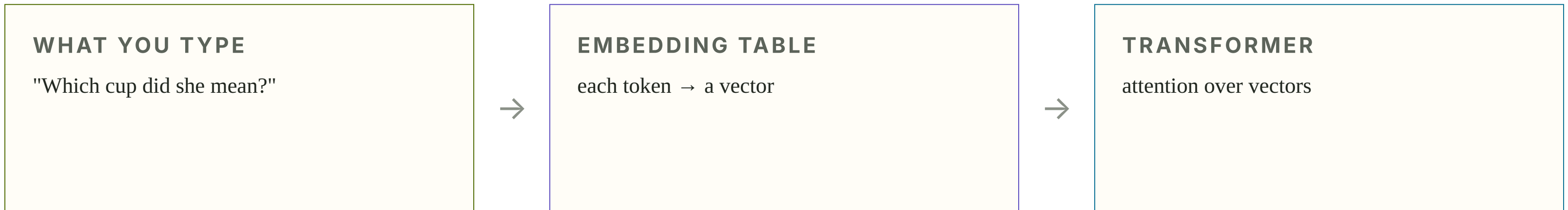
"Which cup did she mean?"



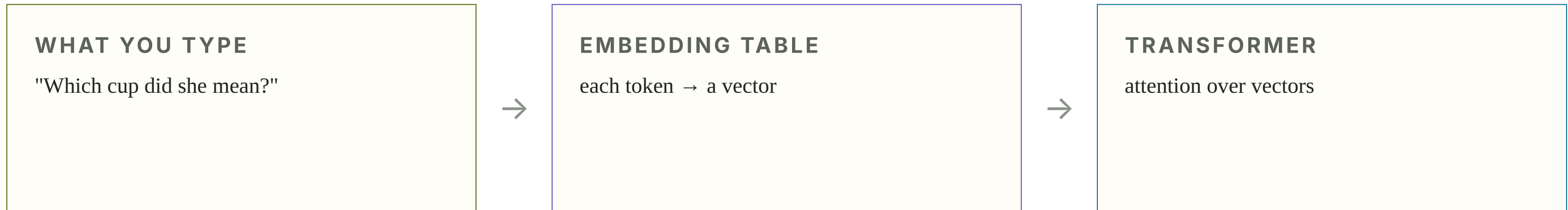
EMBEDDING TABLE

each token $\rightarrow$ a vector

The language model's interface is a sequence of vectors



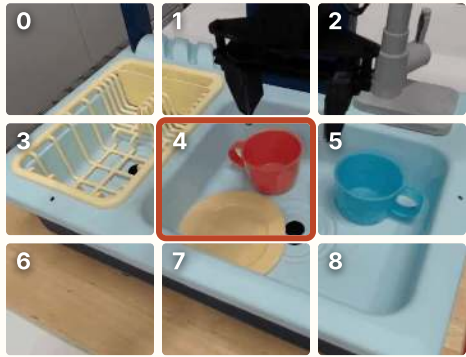
The language model's interface is a sequence of vectors



*Any vector of the right width can enter the sequence.
Nothing requires it to come from text.*

Image patches as tokens (LLaVA)

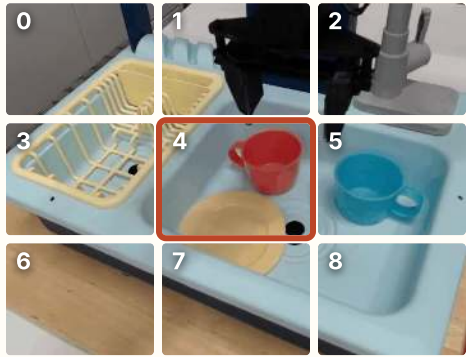
IMAGE · PATCHES 0-8



cut into patches

Image patches as tokens (LLaVA)

IMAGE · PATCHES 0–8



cut into patches



VISION ENCODER



nine patches → nine feature vectors

Image patches as tokens (LLaVA)

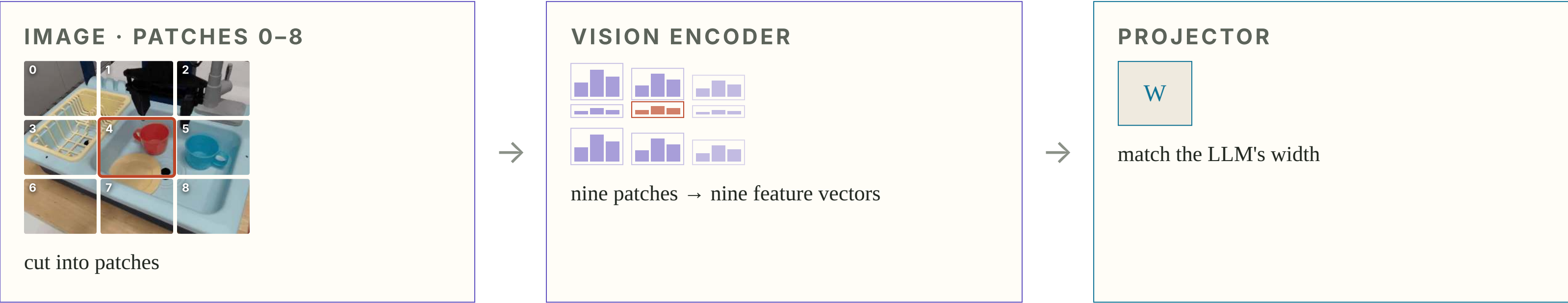


Image patches as tokens (LLaVA)

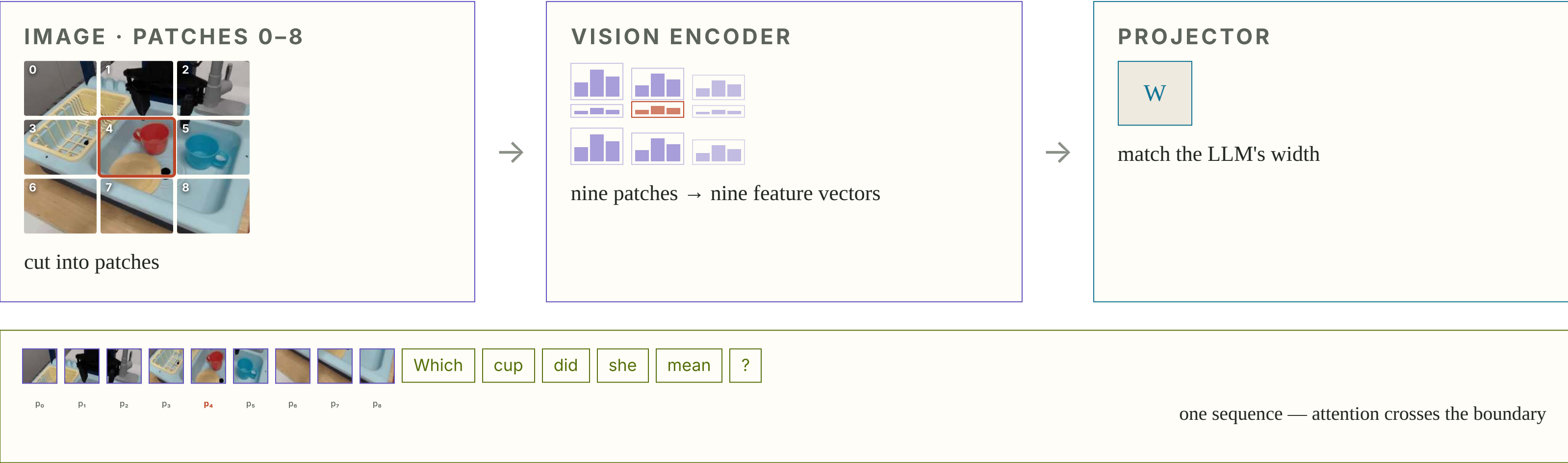
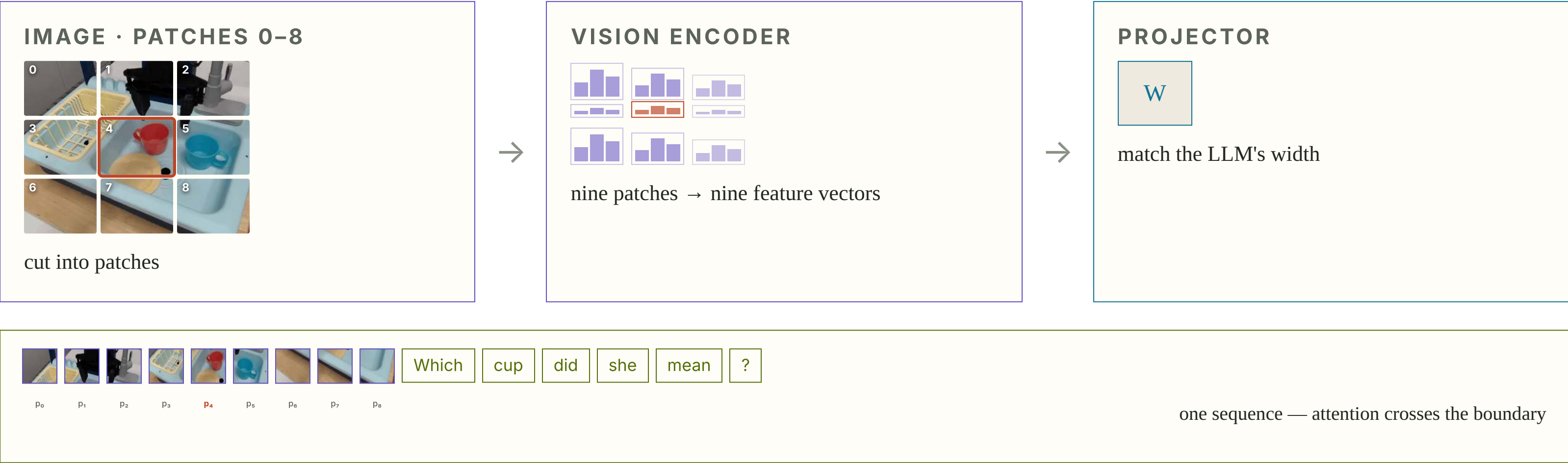


Image patches as tokens (LLaVA)



Patch 4 holds the red cup. Its vector p_4 is how that corner of the world enters the sentence — when the answer says "*the red one*", attention lands on p_4 .

Training the projector: captions, then instructions

STAGE 1 · ALIGNMENT

image + caption → predict the caption

projector learns; big components frozen

Training the projector: captions, then instructions

STAGE 1 · ALIGNMENT

image + caption → predict the caption

projector learns; big components frozen

STAGE 2 · INSTRUCTION TUNING

image + question → predict the answer

curated conversations teach the task

Training the projector: captions, then instructions

STAGE 1 · ALIGNMENT

image + caption → predict the caption

projector learns; big components frozen

Both stages: next-token prediction.

STAGE 2 · INSTRUCTION TUNING

image + question → predict the answer

curated conversations teach the task

Audio: the objective decides what is kept



Audio: the objective decides what is kept



PREDICT THE
TRANSCRIPT

Whisper

keeps the words and who spoke when · lets go of prosody, hesitation, the kettle in the background

Audio: the objective decides what is kept



**PREDICT THE
TRANSCRIPT**

Whisper

keeps the words and who spoke when · lets go of prosody, hesitation, the kettle in the background

MATCH A CAPTION

CLAP

keeps what a caption would name — "a dog barking", "rain" · lets go of exact timing and phonetics

Audio: the objective decides what is kept



PREDICT THE TRANSCRIPT	Whisper	keeps the words and who spoke when · lets go of prosody, hesitation, the kettle in the background
MATCH A CAPTION	CLAP	keeps what a caption would name — "a dog barking", "rain" · lets go of exact timing and phonetics
IDENTIFY MASKED UNITS	wav2vec 2.0	keeps phonetic structure for recognition · lets go of much of speaker and setting

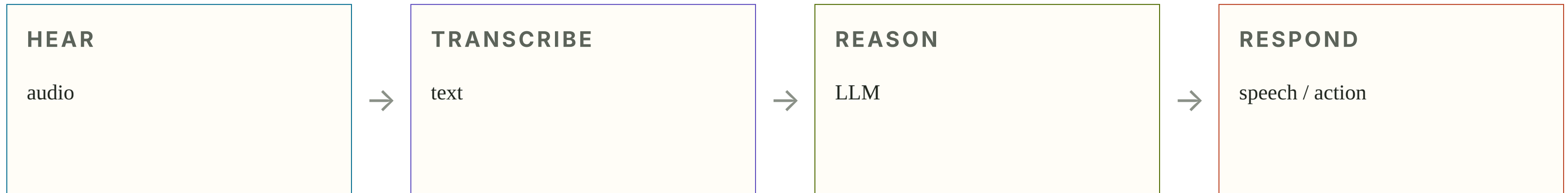
Audio: the objective decides what is kept



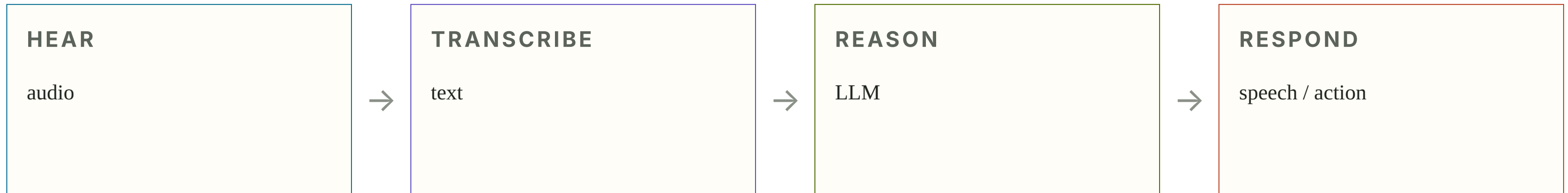
PREDICT THE TRANSCRIPT	Whisper	keeps the words and who spoke when · lets go of prosody, hesitation, the kettle in the background
MATCH A CAPTION	CLAP	keeps what a caption would name — "a dog barking", "rain" · lets go of exact timing and phonetics
IDENTIFY MASKED UNITS	wav2vec 2.0	keeps phonetic structure for recognition · lets go of much of speaker and setting

Same waveform, same door into the language model — a different objective, so a different encoding. (Whisper has an encoder and paired data and is still not contrastive: its loss is next-token prediction.)

The transcript as a bottleneck



The transcript as a bottleneck



SURVIVES

words

speaker turns

MAY BE LOST

hesitation

prosody

overlap

ambient sound

Where multimodal systems are deployed

SEARCH

find by meaning

language → relevant image, video, sound

TRANSCRIBE

audio → text

searchable, inspectable, translatable

INSPECT

ask about a document

charts, screenshots, scenes

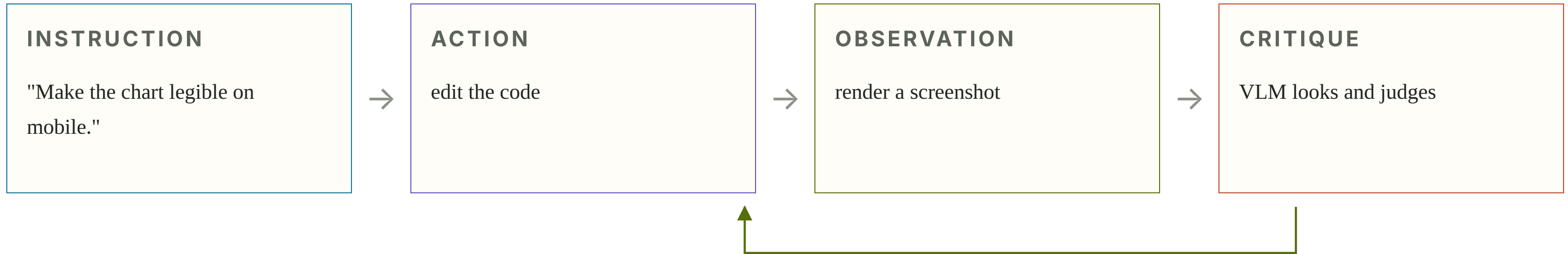
ITERATE

close a digital loop

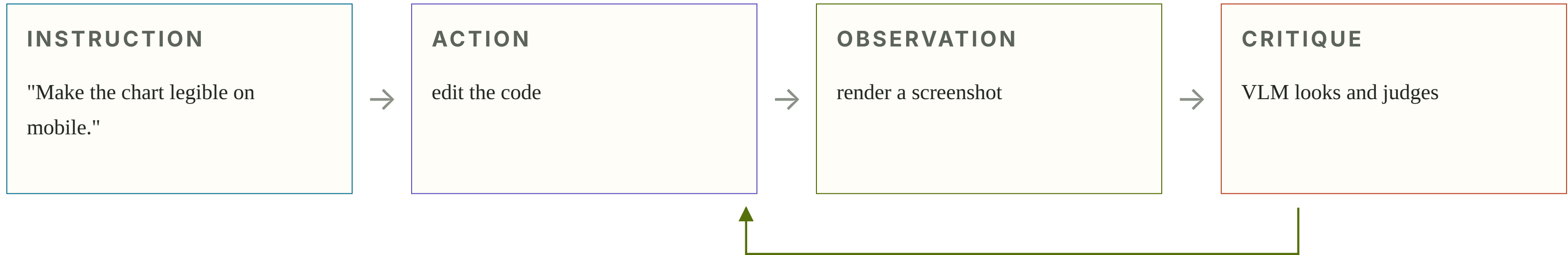
code → render → critique → fix

In all four the output can be inspected and a mistake costs a retry.

A digital control loop: code, render, inspect, fix



A digital control loop: code, render, inspect, fix



Why it works

structured actions · instant feedback · cheap reset · tests exist

Why it still fails

the vision encoder was trained to pair images with text, not to reason about layout — its spatial judgement is not human-like · long-horizon regressions · slow

A correct description is not a plan

THE ANSWER

"The red mug is on the shelf above the kettle."

A correct description is not a plan

THE ANSWER

"The red mug is on the shelf above the kettle."

?

WHAT'S MISSING

handle pose · depth · friction · what happens if you knock it

A correct description is not a plan

THE ANSWER

"The red mug is on the shelf above the kettle."

What changes when the output must move the mug?

?

WHAT'S MISSING

handle pose · depth · friction · what happens if you knock it

Vision—language—action

Language in, action out

OBSERVE → INTERPRET → ACT → OBSERVE AGAIN



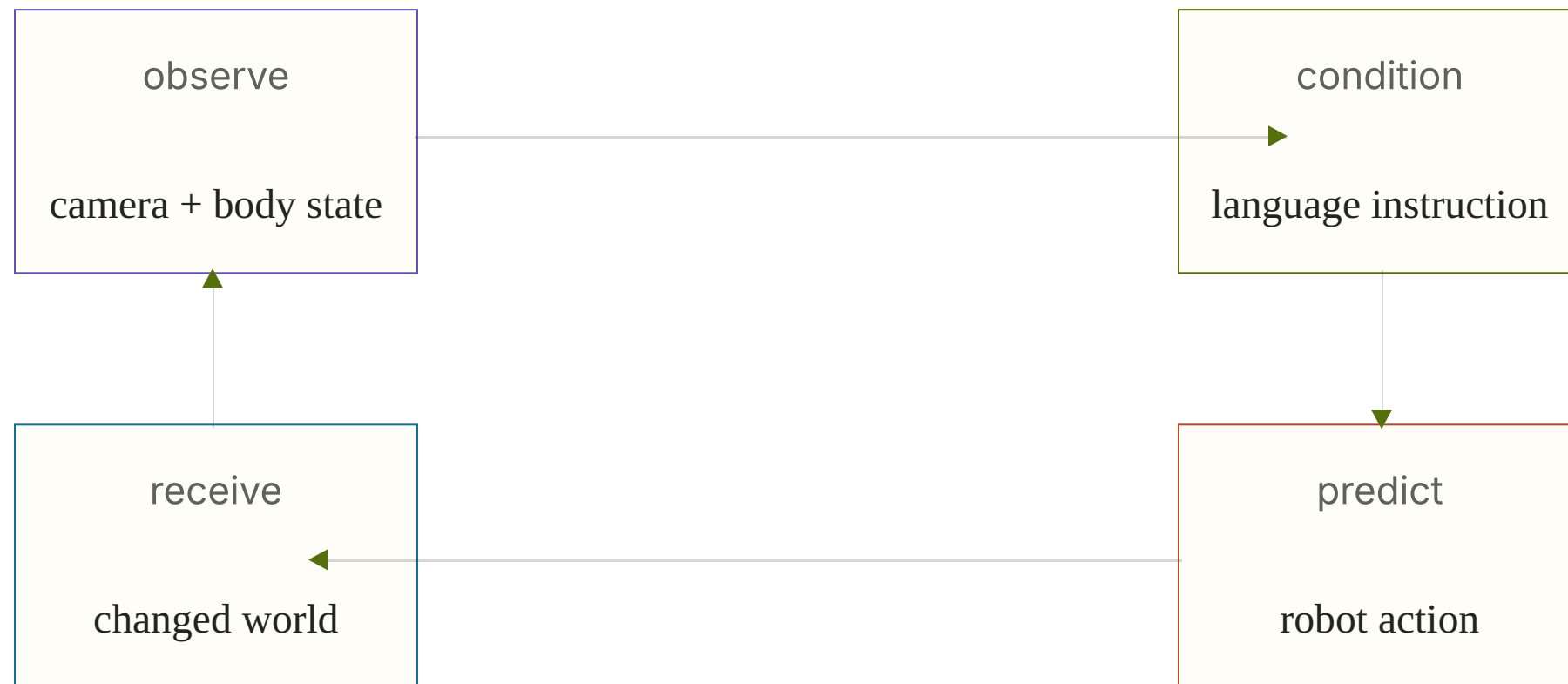
"put the **blue** cup on the plate"



"put the **pink** cup on the plate"

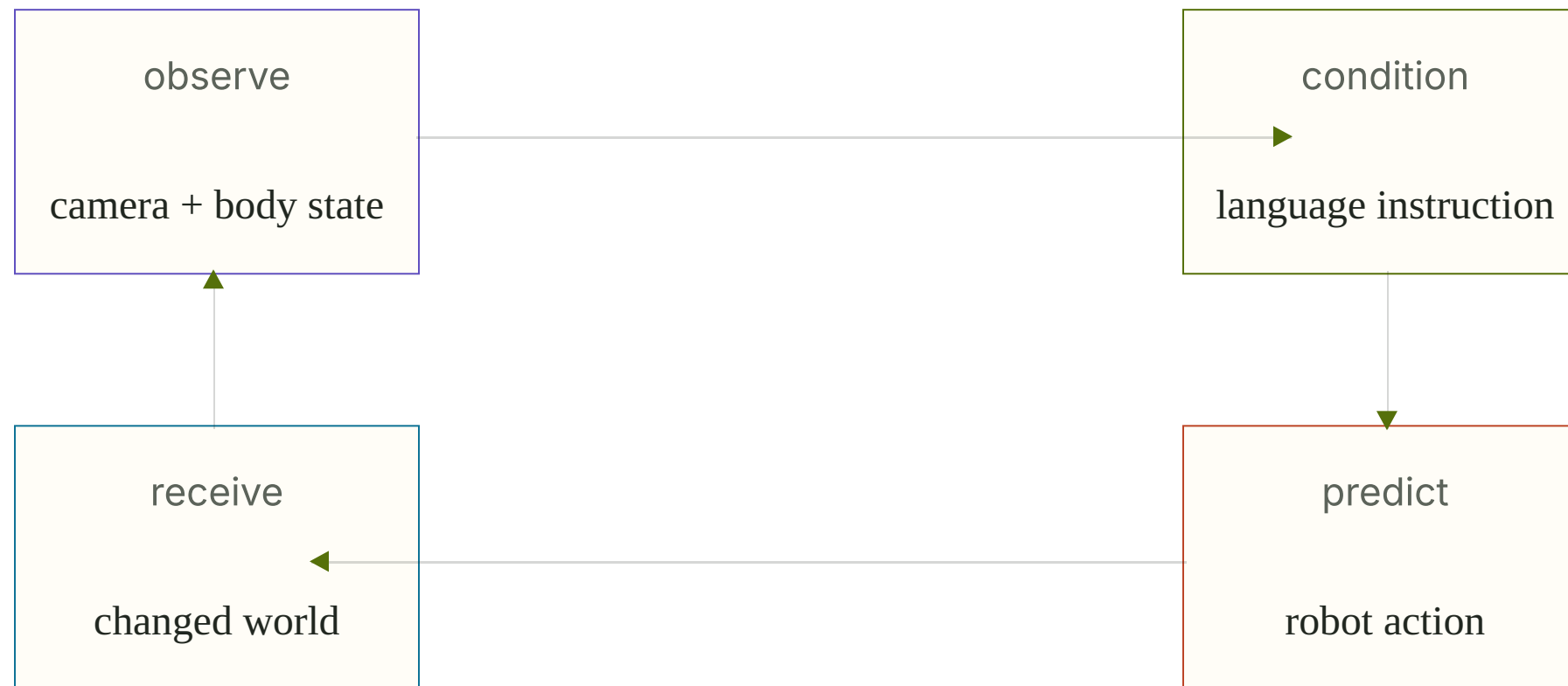
SAME NETWORK, SAME SCENE — ONLY THE INSTRUCTION CHANGES · THERE IS NO PINK CUP — THERE IS A RED ONE, AND RED IS WHAT "PINK" IS NEAR · OPENVLA

The control loop



miss → look → correct · OpenVLA

The control loop



miss → look → correct · OpenVLA

The action changes the next observation.

Where the action labels come from

OBSERVATION

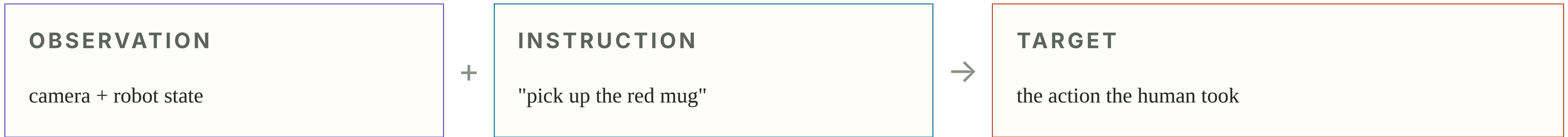
camera + robot state

+

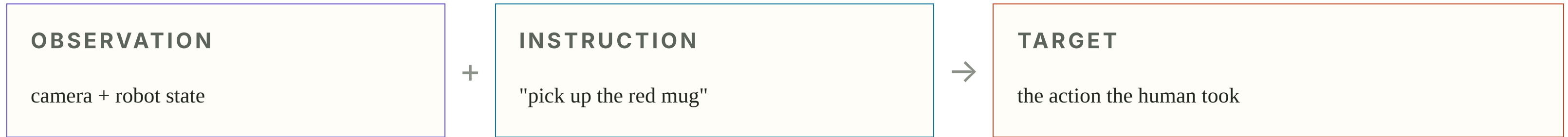
INSTRUCTION

"pick up the red mug"

Where the action labels come from



Where the action labels come from



Behaviour cloning: predict the demonstrator's action from the context they saw.

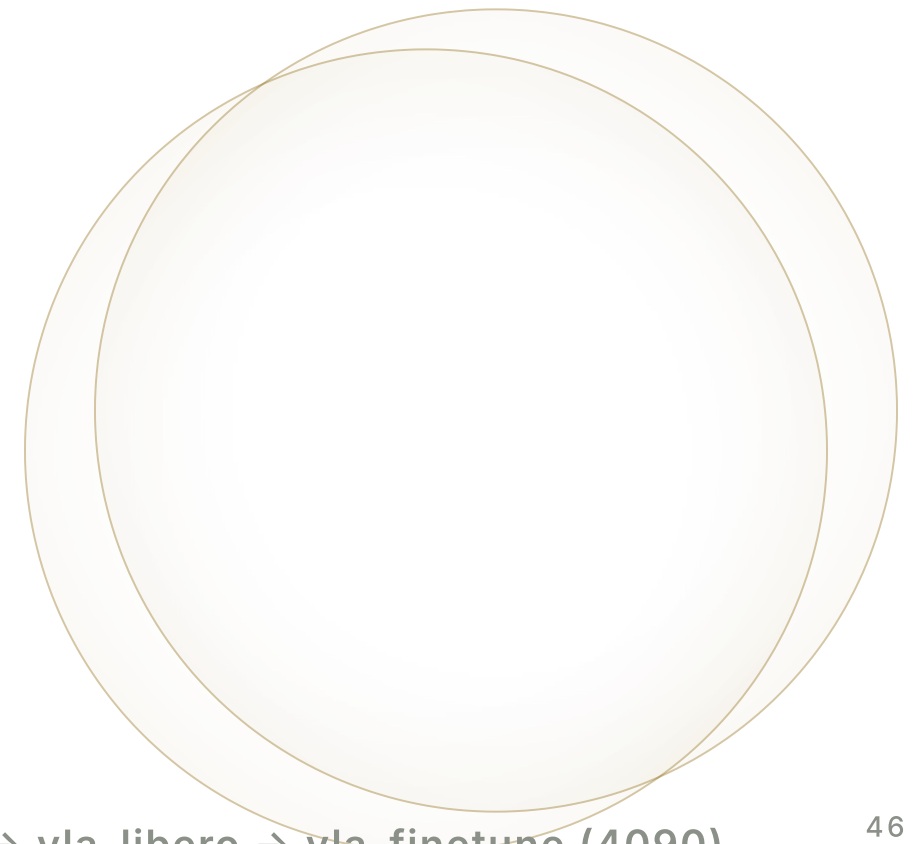
Outside the training data

YOU WILL SEE

a closed-loop rollout, then an instruction-conditioned VLA in simulation

NOTICE

recovery, shortcut behaviour, and where unfamiliar observations enter



Same scene, same weights, four sentences



"pick up the alphabet soup and place it in the basket" — as trained · **done in 123 steps**



"pick up the **tomato sauce** and place it in the basket" · still picks the alphabet soup — the scene decides



"**put** the alphabet soup **into** the basket" · a paraphrase — it never commits to a grasp



"place it **next to** the basket" · it does — and the trained checker still scores **failure**

SMOLVLA FINE-TUNED ON LIBERO · ONE START STATE, ONLY THE SENTENCE DIFFERS · RECORDED FROM DEMO LIBERO ON THE 4090

Why VLA works

WEB PRETRAINING

objects, scenes, language

robot data doesn't have to teach what a mug is

CLOSED LOOP

error correction

act, observe, correct

DEMONSTRATIONS

embodied grounding

what actions look like on this body

LANGUAGE

task interface

one model, many instructions

Why VLA works

WEB PRETRAINING

objects, scenes, language

robot data doesn't have to teach what a mug is

CLOSED LOOP

error correction

act, observe, correct

Semantics transfer. Embodiment-specific details need robot data.

DEMONSTRATIONS

embodied grounding

what actions look like on this body

LANGUAGE

task interface

one model, many instructions

Three action representations

RT-2

action tokens

actions become words
in the vocabulary

OPENVLA

discretised action
tokens

256 bins per action
dimension, like RT-2

π_o

continuous action
expert

flow matching, not
discrete tokens



π_o folding laundry

4x speed

"push the red marker to the
video game controller"

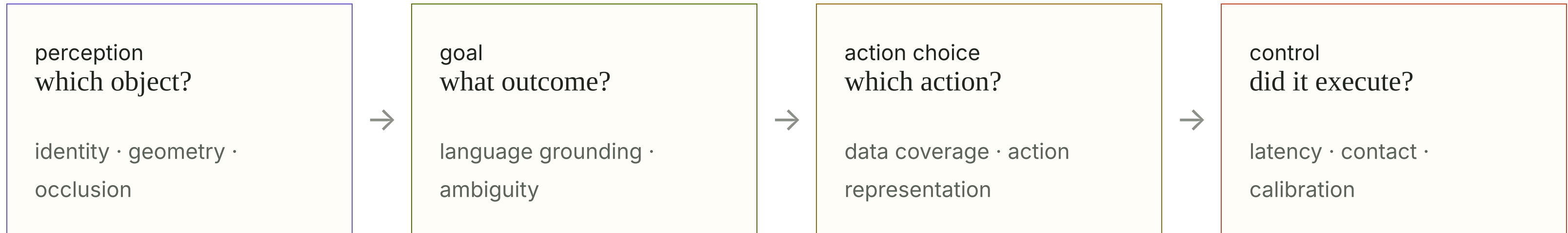
4x speed

Right object. Wrong action.

What did it understand — what's missing from its representation — and where else could failure have entered?

"push the red marker to the
video game controller"

Four places a failure can live



A success rate alone cannot say which one failed.

OpenVLA · succeeds



Octo baseline · fails



Same task. Same instruction. Different model.

Evaluating a robot model

NECESSARY

decomposed tests

object selection · grounding · planning · grasp · recovery

Benchmarks: SimplerEnv (simulation) · LIBERO · Open X-Embodiment evaluations. Each reports a success rate; ask which of these it decomposes.

NOT SUFFICIENT

one success percentage

hides task mix, camera setup, embodiment, retries, human intervention

Evaluating a robot model

NECESSARY

decomposed tests

object selection · grounding · planning · grasp · recovery

Benchmarks: SimplerEnv (simulation) · LIBERO · Open X-Embodiment evaluations. Each reports a success rate; ask which of these it decomposes.

Can the evaluation tell you why the model succeeded?

NOT SUFFICIENT

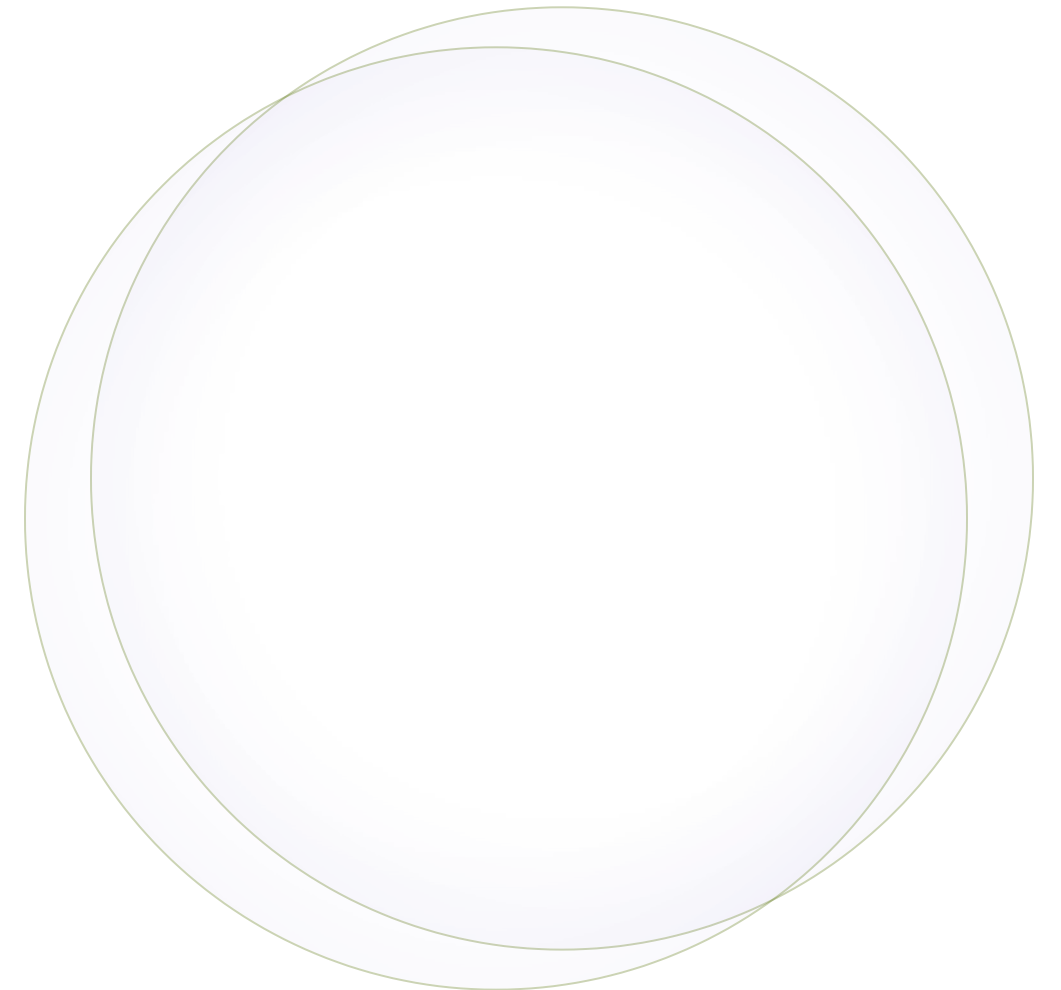
one success percentage

hides task mix, camera setup, embodiment, retries, human intervention

Predict before acting?

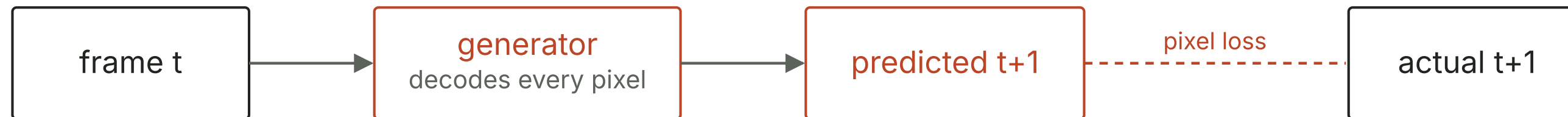
World models and planning

STATE + CANDIDATE ACTION → PREDICTED CONSEQUENCE



Pixel prediction vs latent prediction

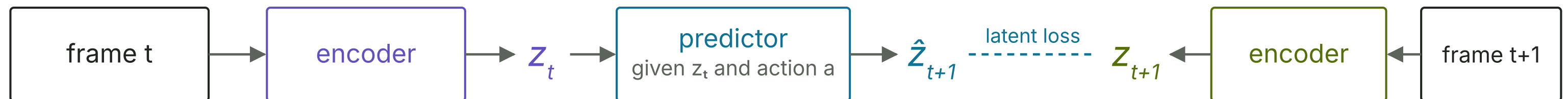
PIXEL PREDICTION



this does not work as a representation objective

capacity goes on texture, shadow and reflection; predictions blur to the mean; nothing forces the features that decide the action

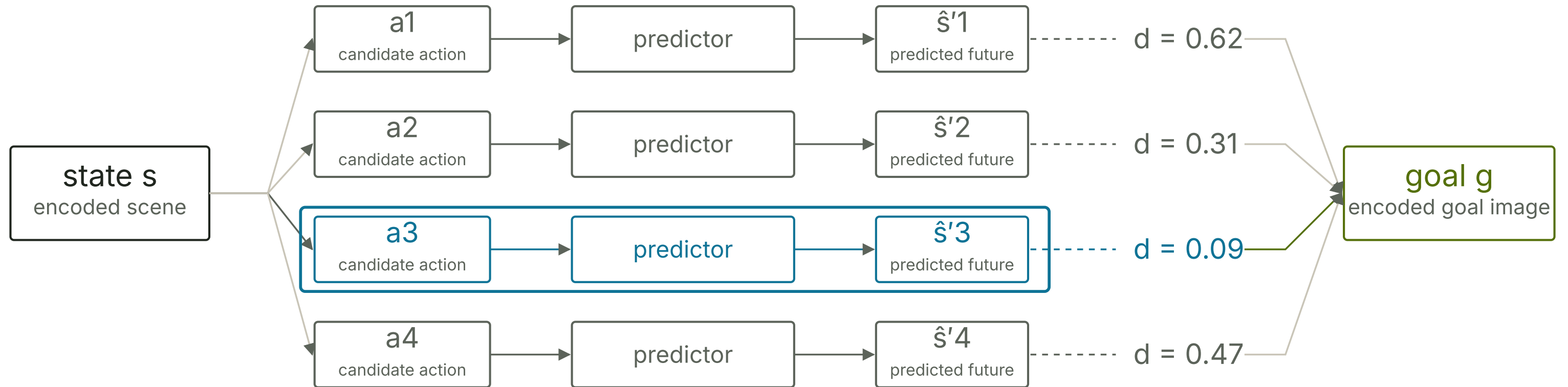
LATENT PREDICTION



only what the encoder keeps counts: two futures that differ in something the encoder dropped are the same future

the encoder decides which futures count as different — the funnel, chosen on purpose (I-JEPA, V-JEPA)

An action-conditioned world model



score a few hundred candidates, execute the one whose predicted future lands nearest the goal, observe, repeat
model-predictive control in a learned latent space — the expectation is exposed before acting; the price is search on every step

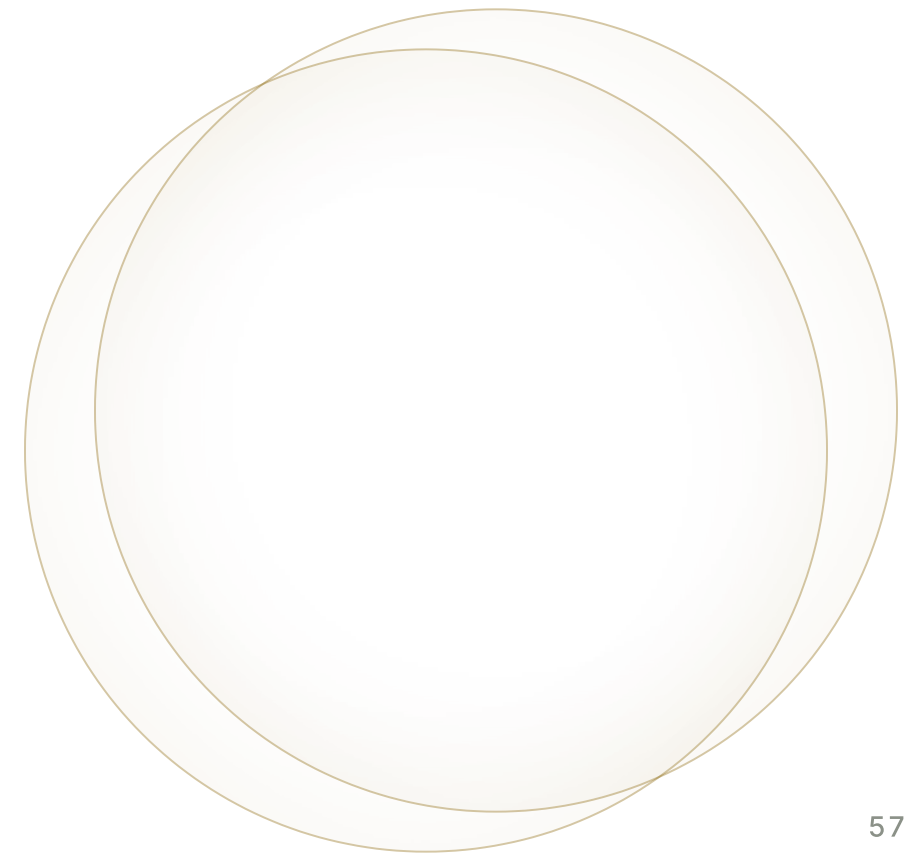
Score futures before acting

YOU WILL SEE

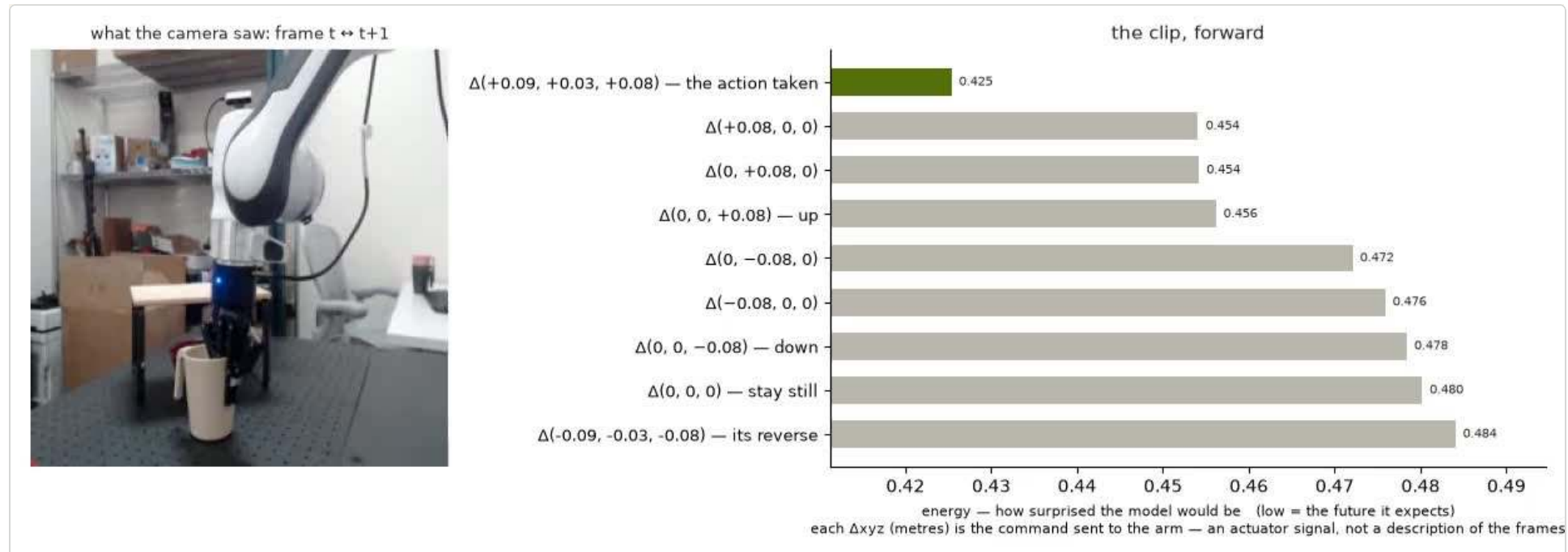
V-JEPA 2-AC assign an energy to candidate robot actions

NOTICE

the chosen action emerges from a landscape, not from one forward pass

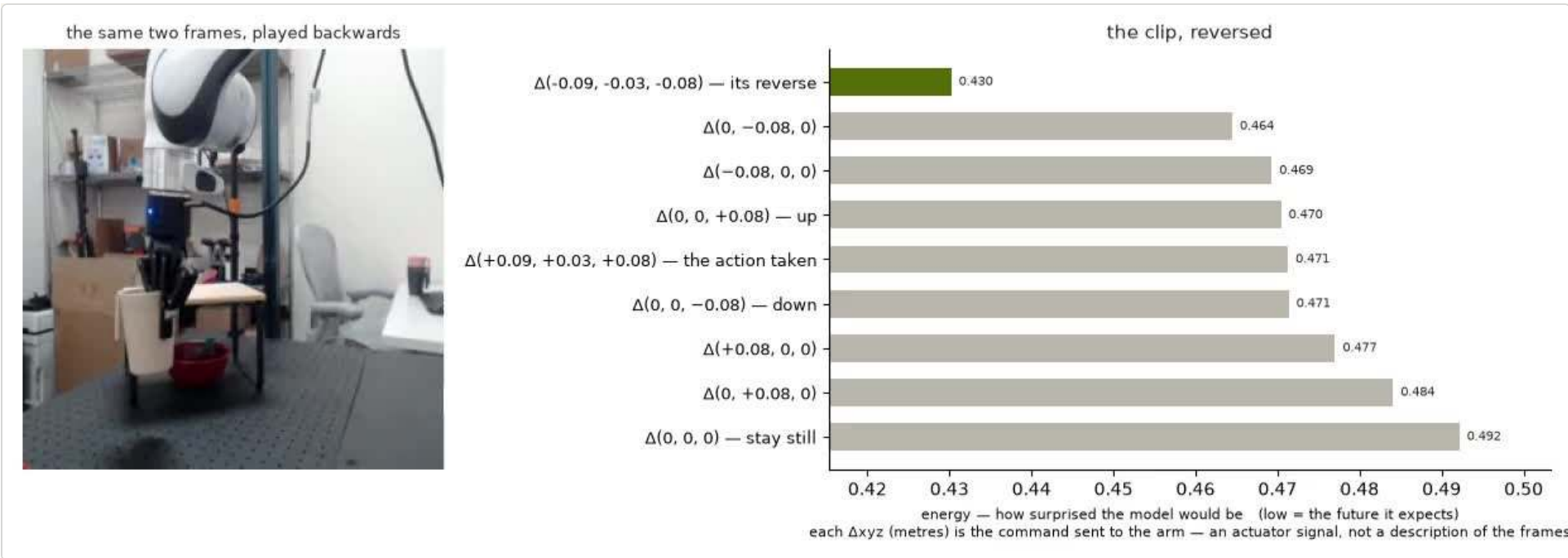


Nine actions, three clips



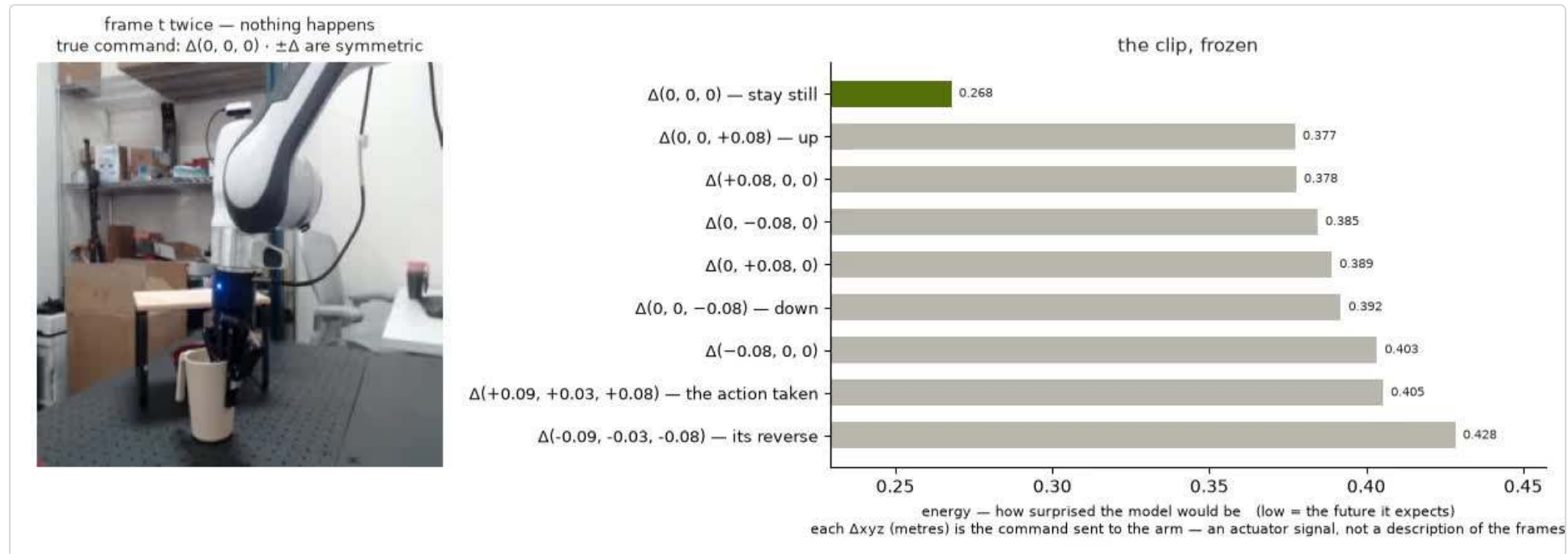
each bar is a Δxyz **command sent to the arm** — an actuator signal, not a description of the frames — scored against the clip as it happened: **the taken action wins**, and commands sharing its components beat their opposites

Nine actions, three clips



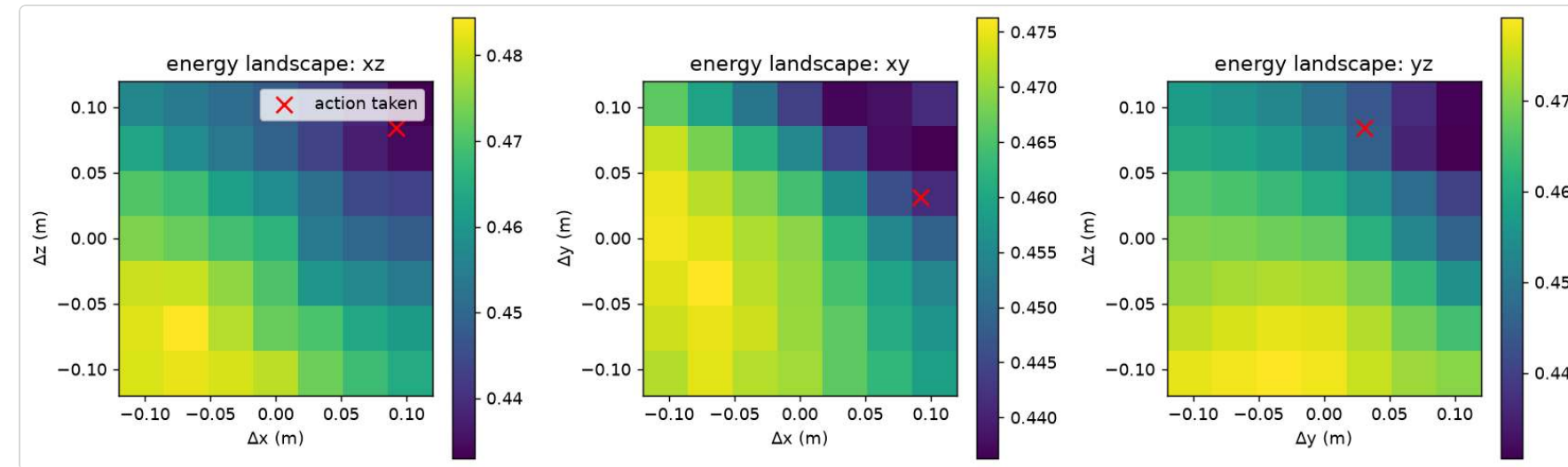
press: the same two frames, played backwards — **its reverse wins**, and the taken action drops mid-field

Nine actions, three clips



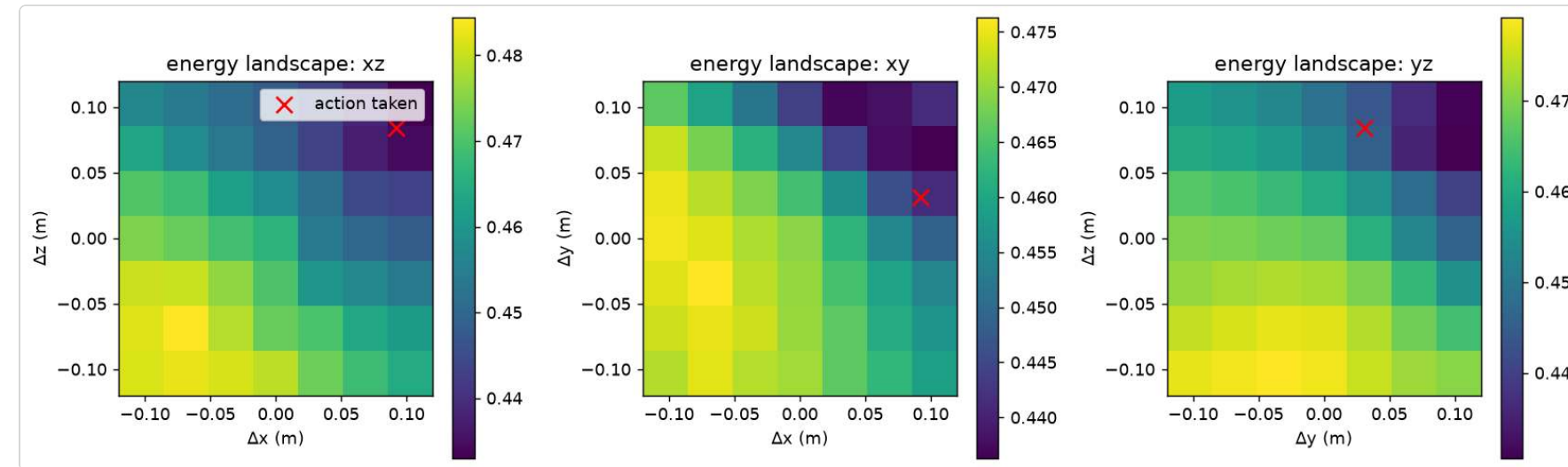
press: frame t twice — the true command is now $\Delta(0, 0, 0)$, and **"stay still" wins**; $+\Delta$ and $-\Delta$ become symmetric guesses and land together

The whole grid

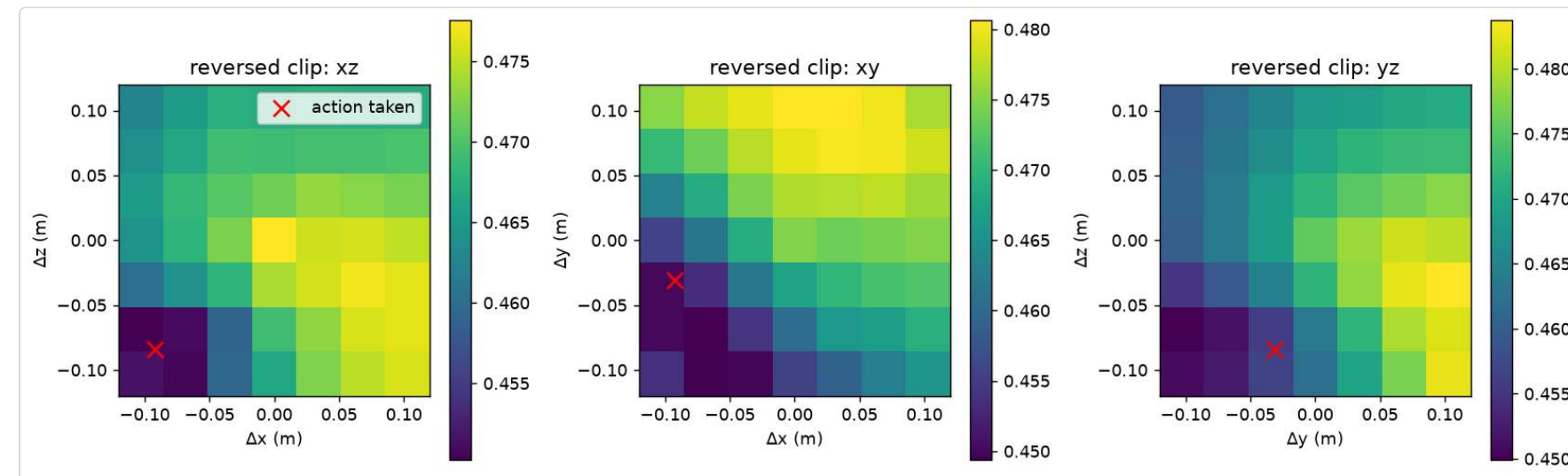


now every candidate at once: 343 Δxyz on a 12 cm grid — the minimum lands beside the action the arm actually took (✕)

The whole grid

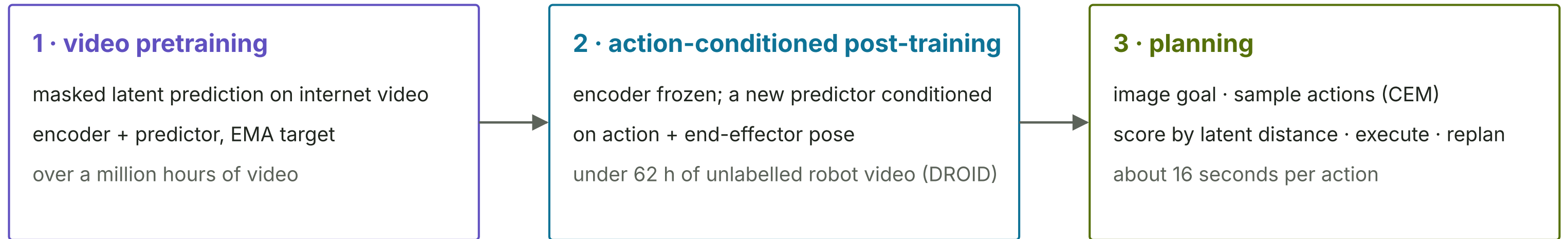


now every candidate at once: 343 Δxyz on a 12 cm grid — the minimum lands beside the action the arm actually took (✕)



the same control at grid scale: reverse the clip and the minimum **flips with it**

V-JEPA 2: what it shows and what it does not



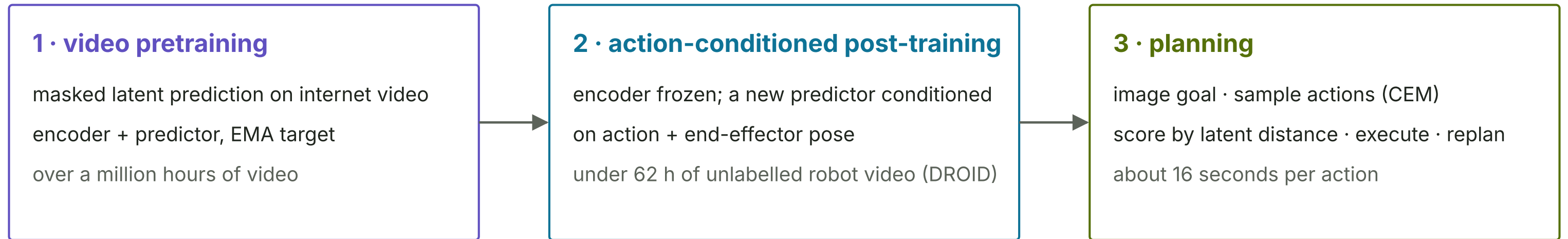
V-JEPA 2 · Assran et al. 2025 — the representation comes from stage 1; the robot data only teaches the predictor how actions move it

WHAT THE WORK SHOWS

video pretraining can support robot planning

action-conditioned post-training · image-goal tasks · real robot

V-JEPA 2: what it shows and what it does not



V-JEPA 2 · Assran et al. 2025 — the representation comes from stage 1; the robot data only teaches the predictor how actions move it

WHAT THE WORK SHOWS

video pretraining can support robot planning

action-conditioned post-training · image-goal tasks · real robot

WHAT IT DOESN'T SHOW

fast, language-conditioned general control

~16 seconds per action · image goals only · camera-sensitive · horizon errors

VLA vs world-model planner

VLA

fast amortised action selection

language interface · demonstration support · silent failure outside training distribution

WORLD MODEL PLANNER

explicit consequence prediction

online search · model error · planning cost · usually image goals

VLA vs world-model planner

VLA

fast amortised action selection

language interface · demonstration support · silent failure outside training distribution

WORLD MODEL PLANNER

explicit consequence prediction

online search · model error · planning cost · usually image goals

VLM specifies goal



a VLA proposes actions



world model evaluates consequences



controller executes

Why was the ordinary request harder?

Why was the ordinary request harder?

SOLAR SYSTEM MODEL

The task never leaves text and code: run it, check it, fix it. Every retry is free.

Why was the ordinary request harder?

SOLAR SYSTEM MODEL

The task never leaves text and code: run it, check it, fix it. Every retry is free.

A CUP OF TEA

The task crosses from words to pixels to a body acting on the world. Each action changes what the camera sees next, and a wrong one is not undone.

Why was the ordinary request harder?

SOLAR SYSTEM MODEL

The task never leaves text and code: run it, check it, fix it. Every retry is free.

A CUP OF TEA

The task crosses from words to pixels to a body acting on the world. Each action changes what the camera sees next, and a wrong one is not undone.

predict the next word → relate signals → answer → act → predict consequences

Moravec's paradox, 1988 — now with the mechanism.

A representation is not the world.

It's a bet about which differences will matter later.