Evolution of Representations in the Transformer

Lena Voita

Based on EMNLP 2019 paper by Elena Voita[1,2,3] , Rico Sennrich[4,2], Ivan Titov[2,3]

Yandex Research

THE UNIVERSITY of EDINBURGH

UNIVERSITY OF AMSTERDAM

University of Zurich UZH

Words -> words in context

- Shift from static embeddings to contextualized word representations

Words -> words in context

- Shift from static embeddings to contextualized word representations

ELMo

Architecture: bi-LSTM

Training objective: LM

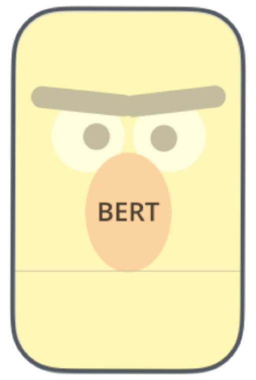
How: add ELMo representations
to the task-specific model

Words -> words in context

- Shift from static embeddings to contextualized word representations

ELMo

BERT



Architecture: bi-LSTM

Training objective: LM

How: add ELMo representations
to the task-specific model

Architecture: Transformer

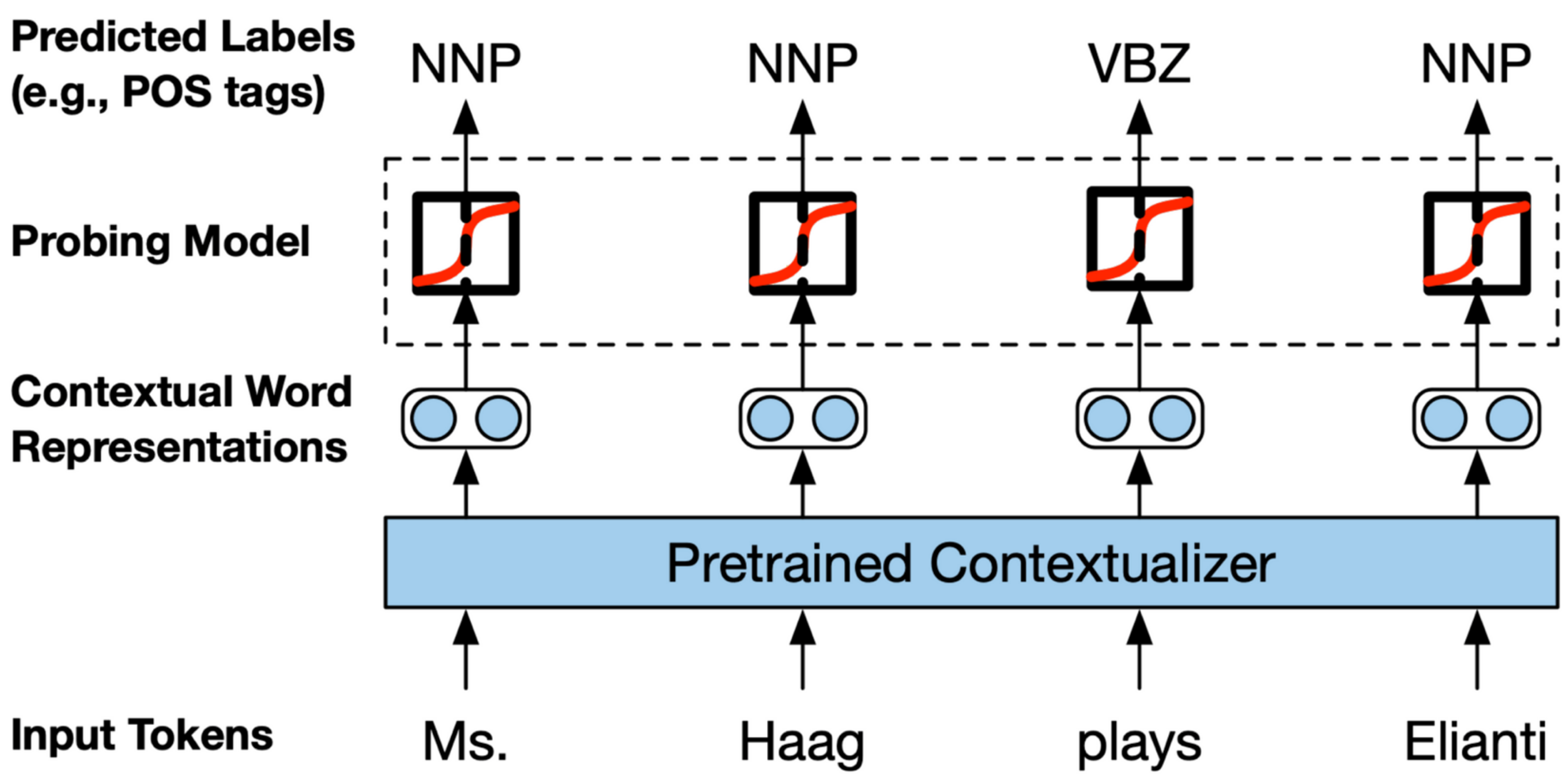
Training objective: MLM

How: use BERT representations
INSTEAD of the task-specific model

And it was the beginning of a very long story…

What do models learn?

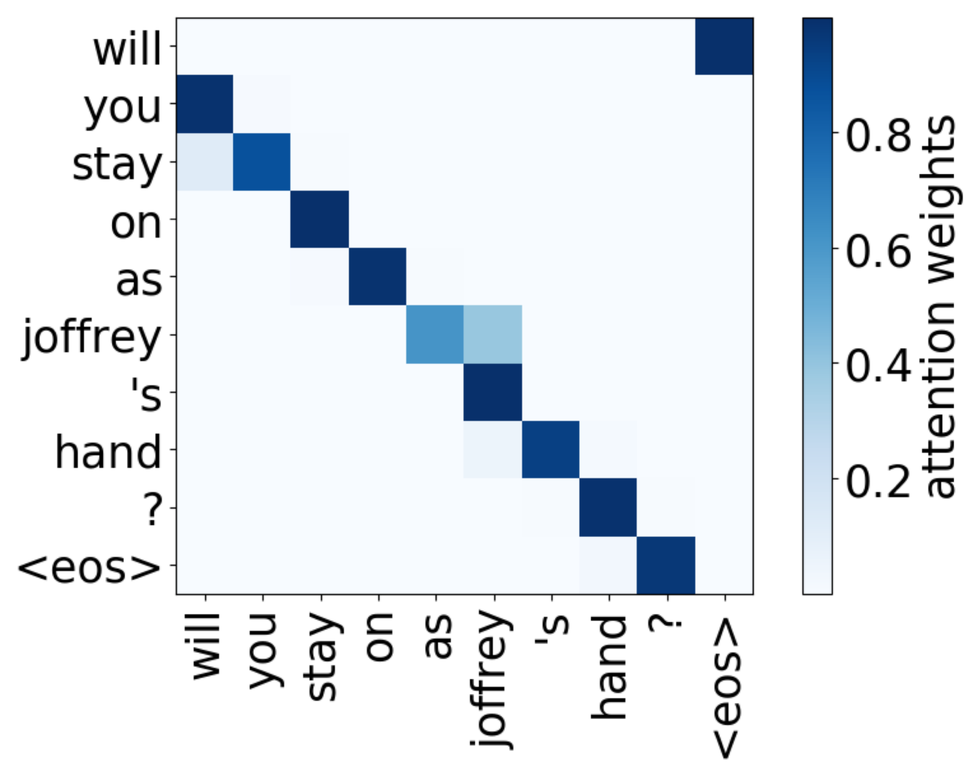
- probing classifiers

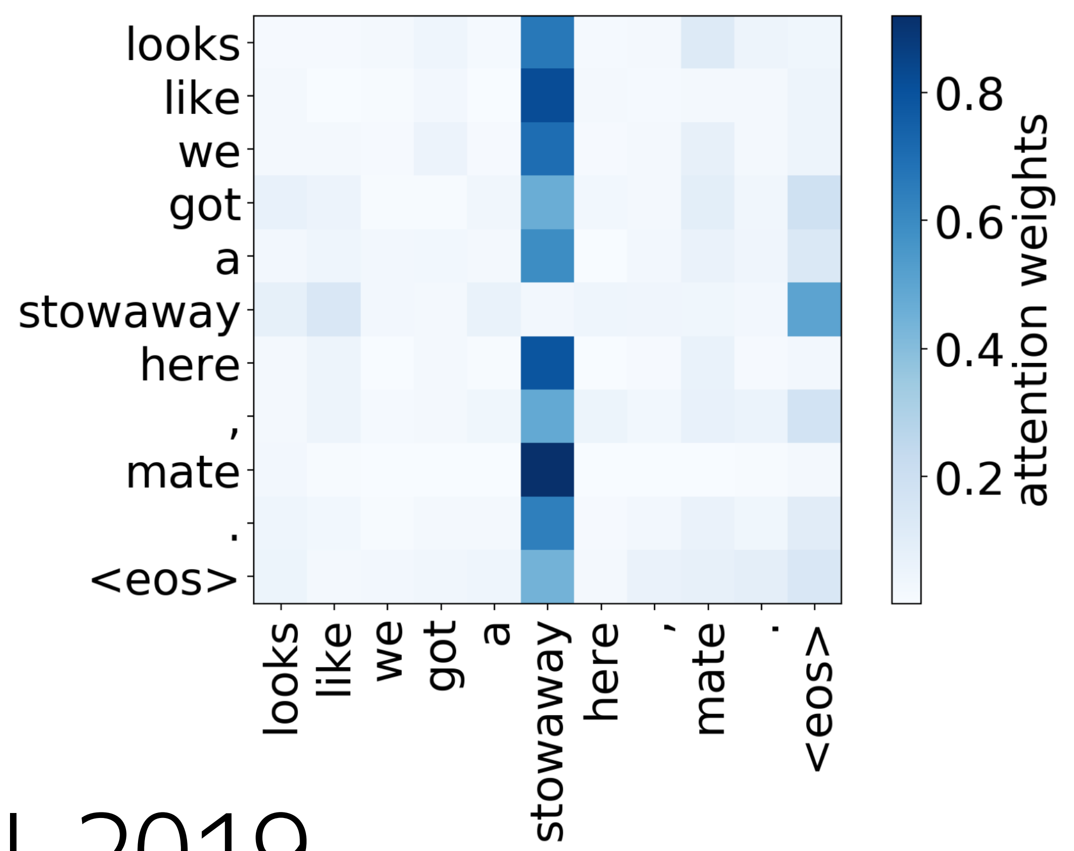


Picture credit: Liu et al, 2019

What do models learn?

- probing classifiers
- model components (e.g., importance and functions of attention heads)





Voita et al, 2019

What do models learn?

- probing classifiers
- model components (e.g., importance and functions of attention heads)
- fill in the blanks

Prompts

DirectX *is developed by* y_{man}

y_{mine} *released the* DirectX

DirectX *is created by* y_{para}

Top 5 predictions and log probabilities

	y_{man}		y_{mine}		y_{para}	
1	Intel	-1.06	Microsoft	-1.77	Microsoft	-2.23
2	Microsoft	-2.21	They	-2.43	Intel	-2.30
3	IBM	-2.76	It	-2.80	default	-2.96
4	Google	-3.40	Sega	-3.01	Apple	-3.44
5	Nokia	-3.58	Sony	-3.19	Google	-3.45

Picture credit: Jiang et al, 2019

Why a more general understanding is important?

It can:

- give intuition for creating a better training objective
- give intuition of how to properly use pretrained representations
- explain “puzzles” from previous work

Plan

- Evolution of representations of individual tokens
- Training objectives: LM, MLM, MT
- ”Puzzles” from previous work
- The Information-Bottleneck: our point of view
- Experiments

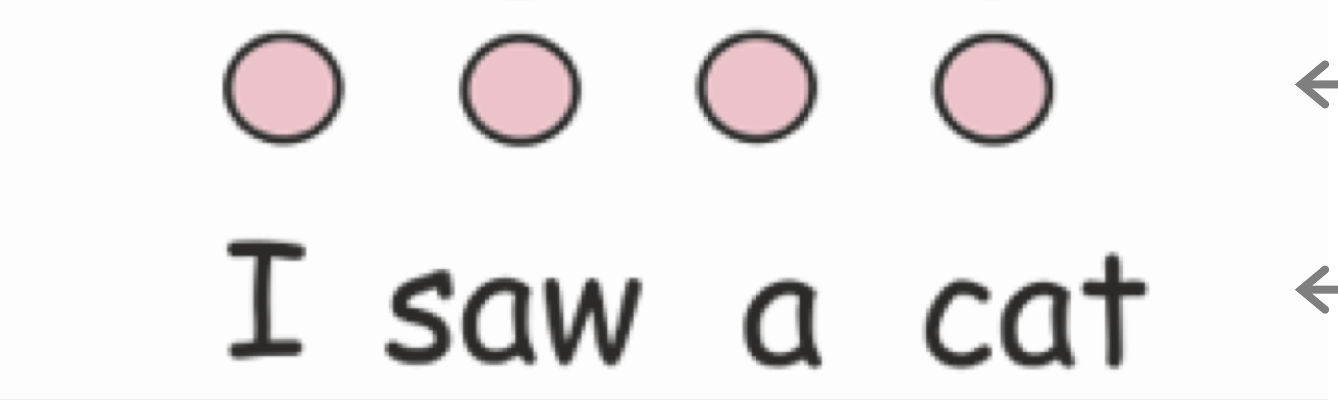
Plan

- Evolution of representations of individual tokens
- Training objectives: LM, MLM, MT
- ”Puzzles” from previous work
- The Information-Bottleneck: our point of view
- Experiments

Representations of individual tokens

I saw a cat ← Input data

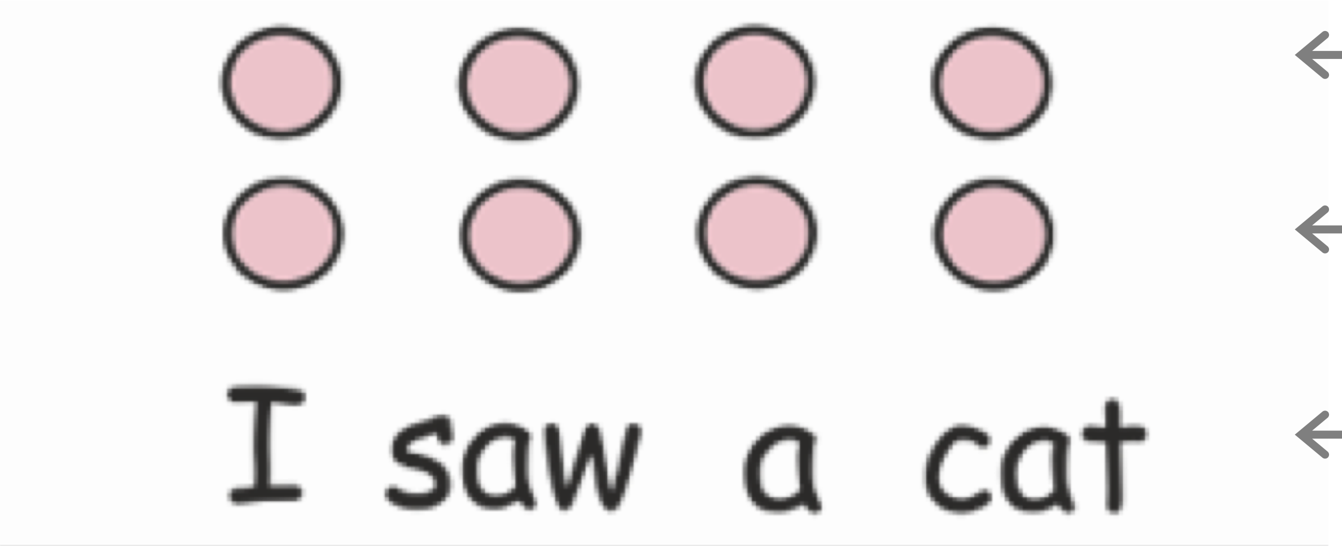
Representations of individual tokens



Initial representations: token and position

Input data

Representations of individual tokens

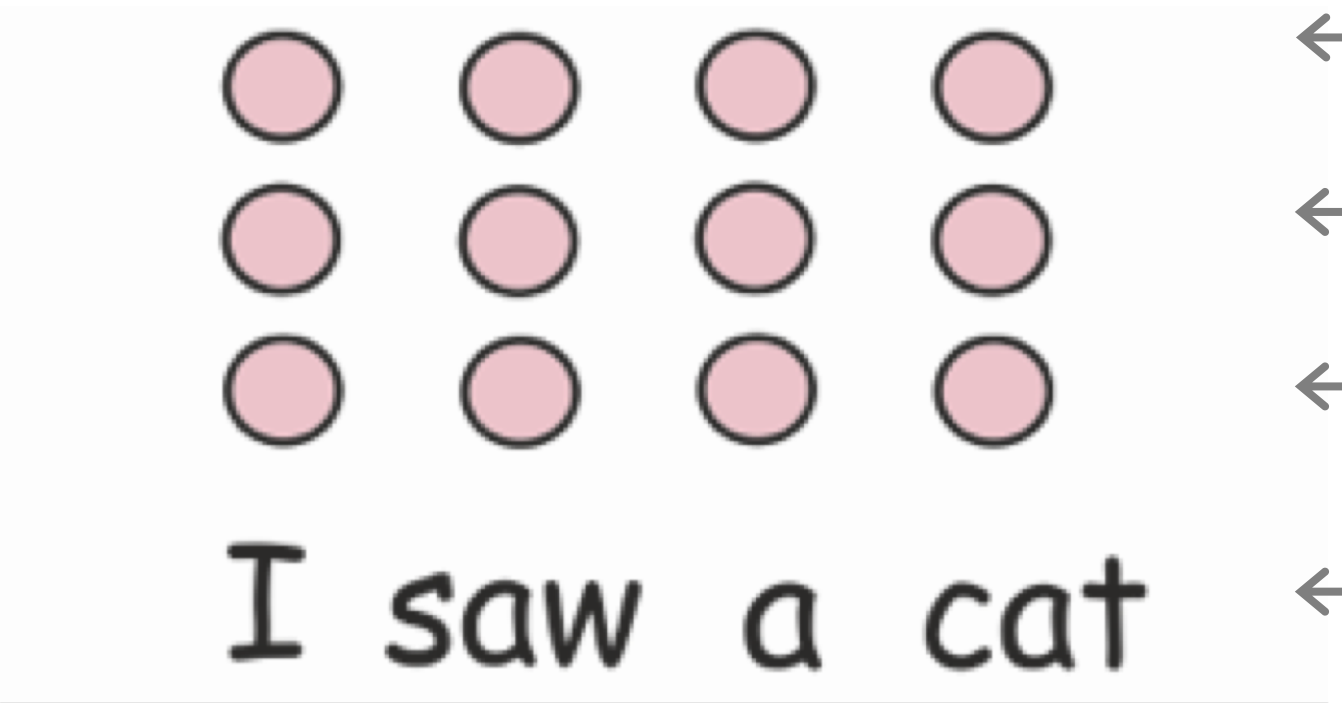


1st layer representations

Initial representations: token and position

Input data

Representations of individual tokens



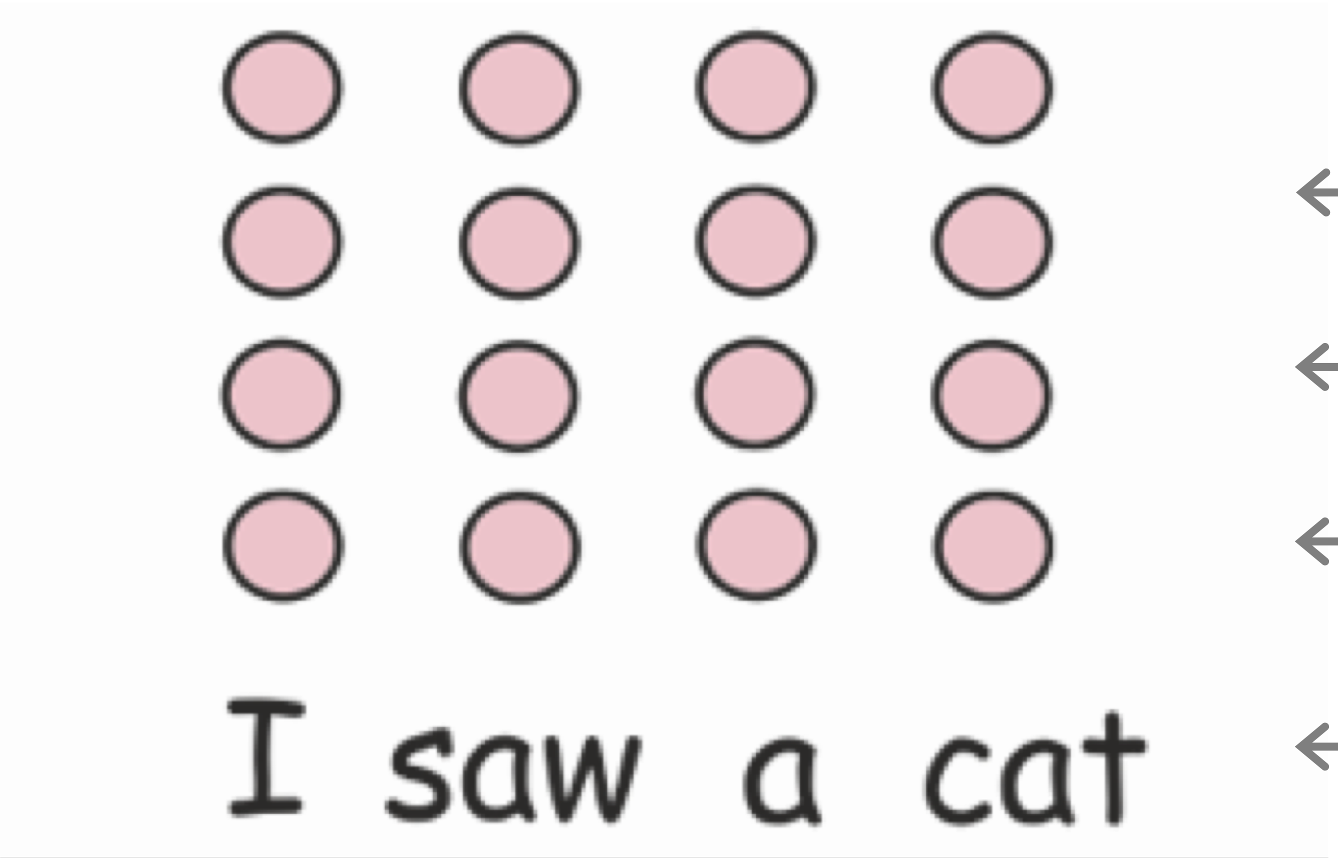
2^{nd} layer representations

1^{st} layer representations

Initial representations: token and position

Input data

Representations of individual tokens



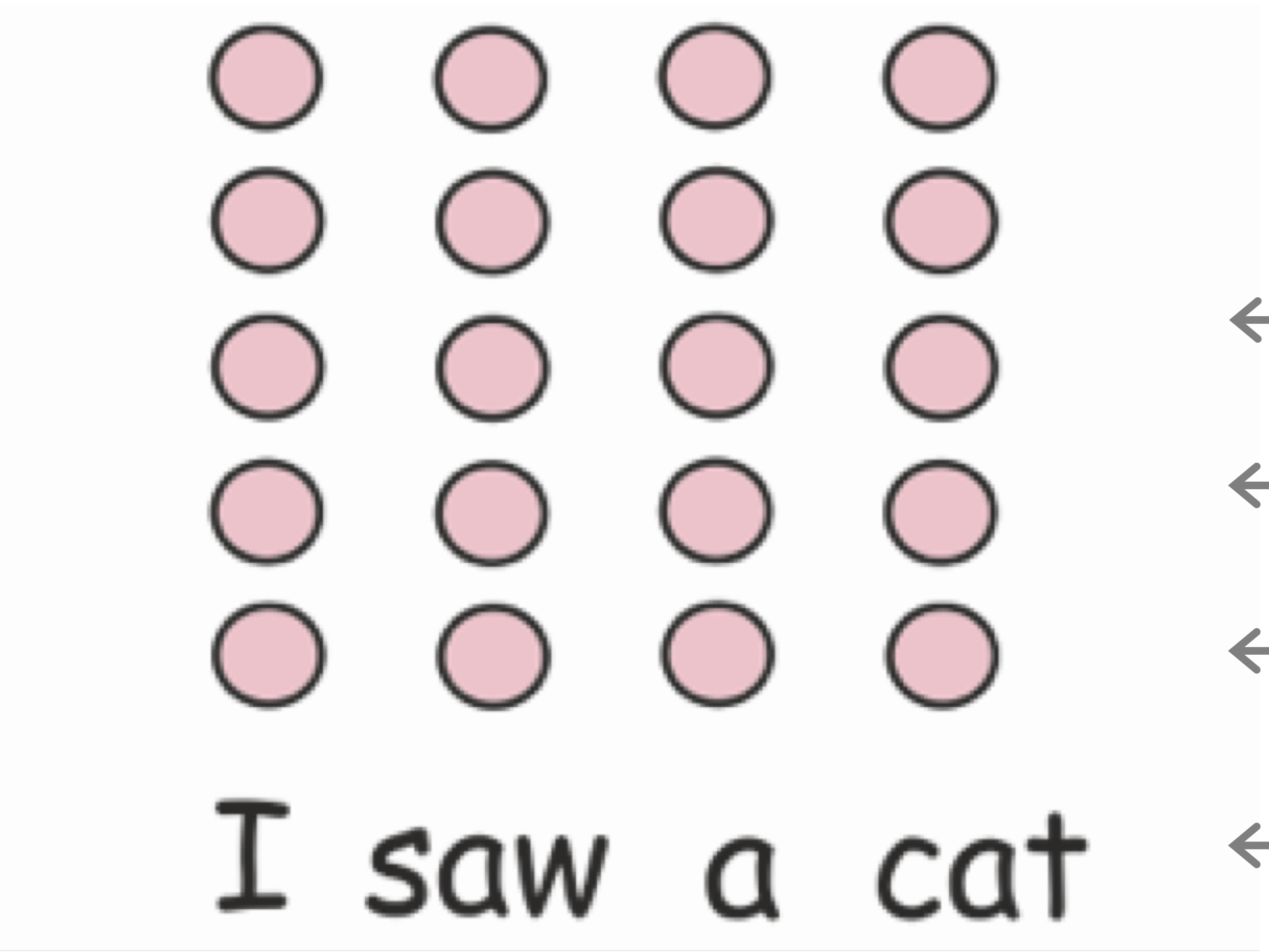
2nd layer representations

1st layer representations

Initial representations: token and position

Input data

Representations of individual tokens



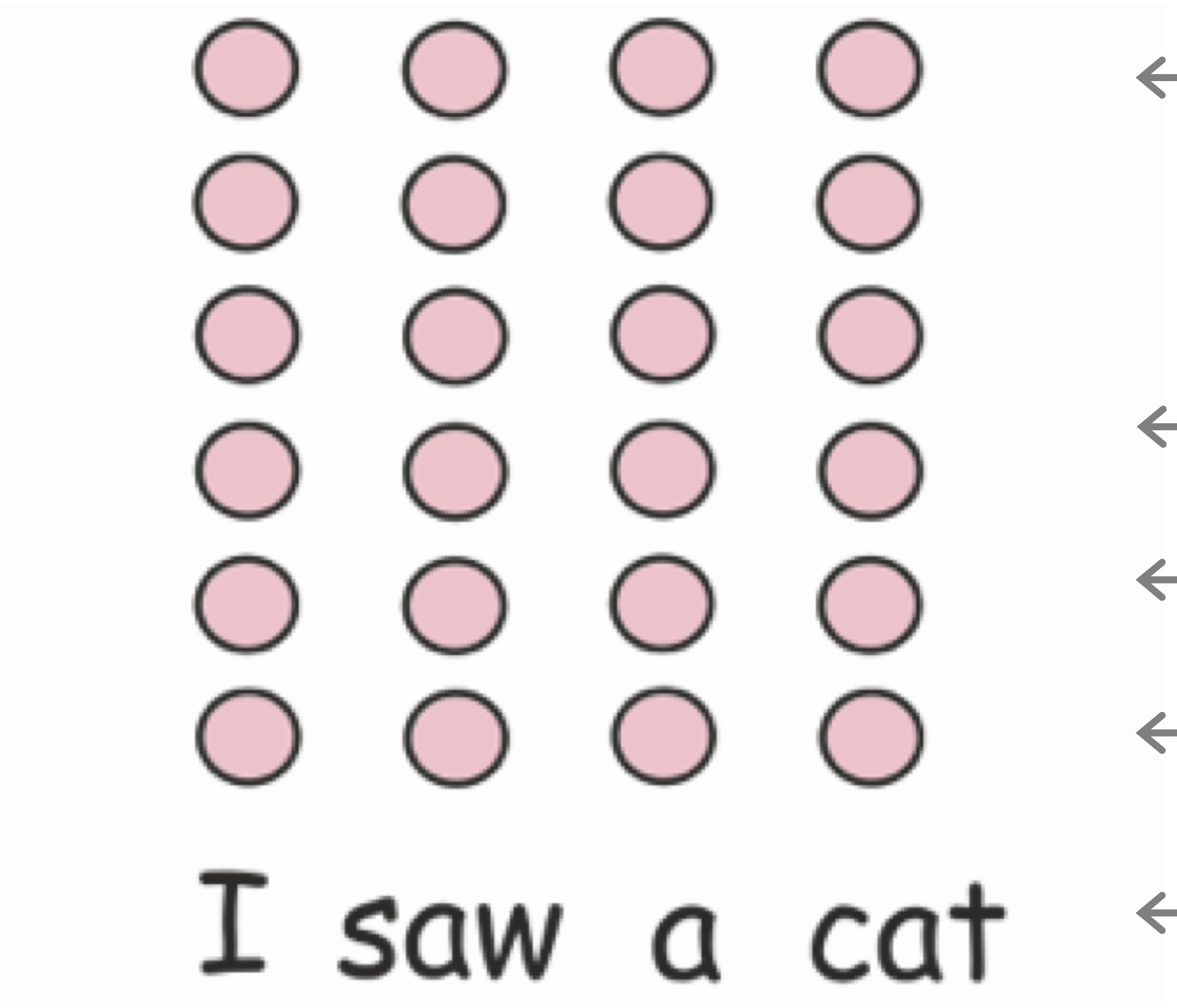
2nd layer representations

1st layer representations

Initial representations: token and position

Input data

Representations of individual tokens



Final representations of tokens

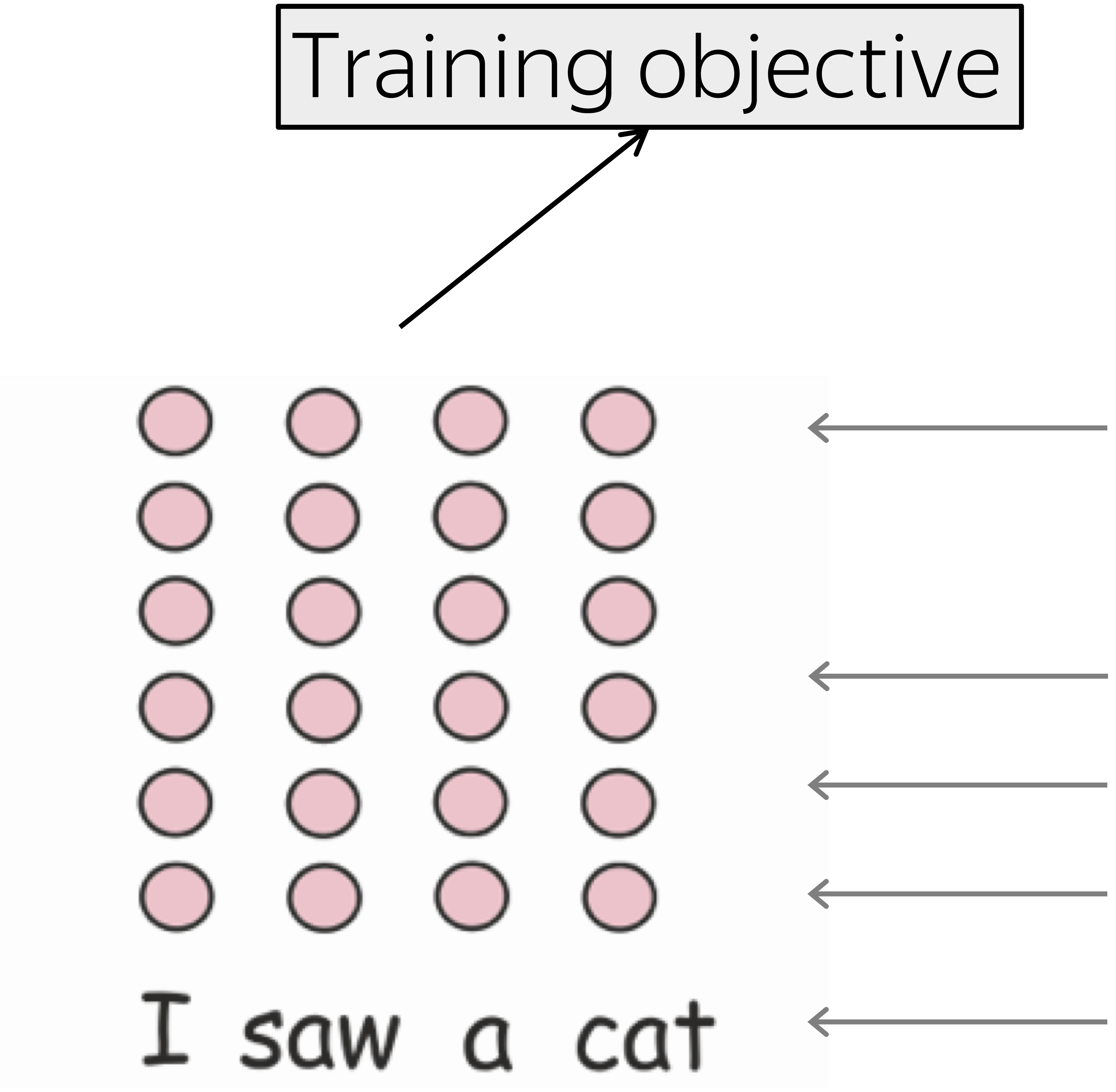
2nd layer representations

1st layer representations

Initial representations: token and position

Input data

Representations of individual tokens



Final representations of tokens

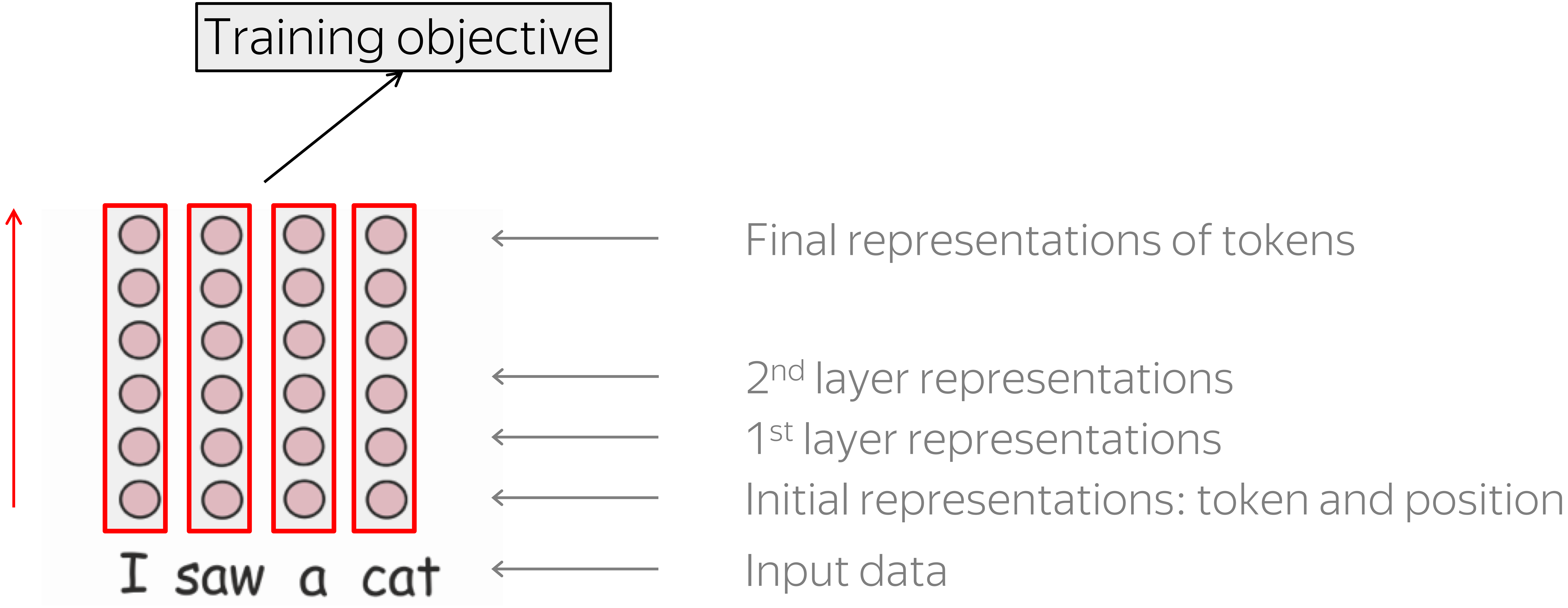
2nd layer representations

1st layer representations

Initial representations: token and position

Input data

Representations of individual tokens



Plan

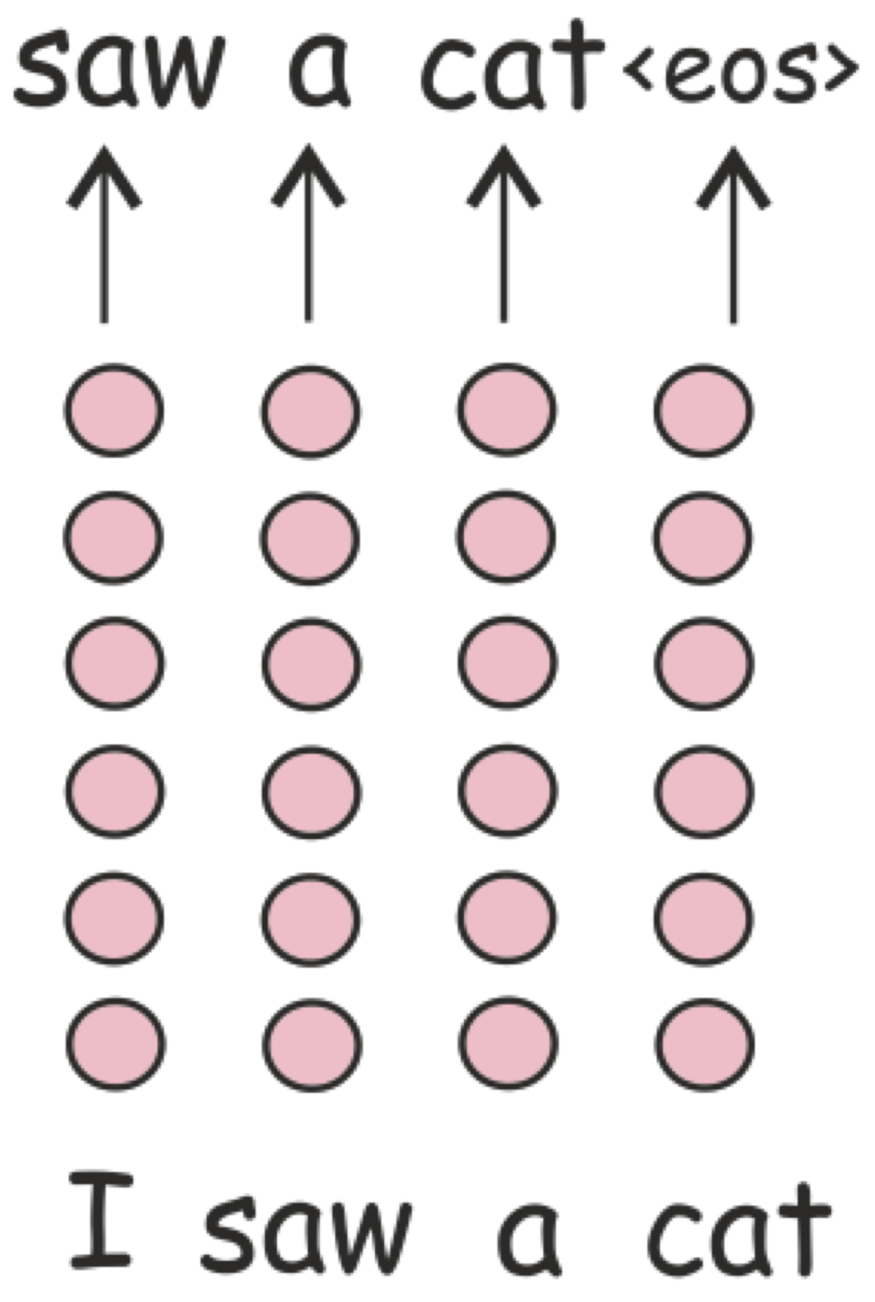
- Evolution of representations of individual tokens
- Training objectives: LM, MLM, MT
- ”Puzzles” from previous work
- The Information-Bottleneck: our point of view
- Experiments

Plan

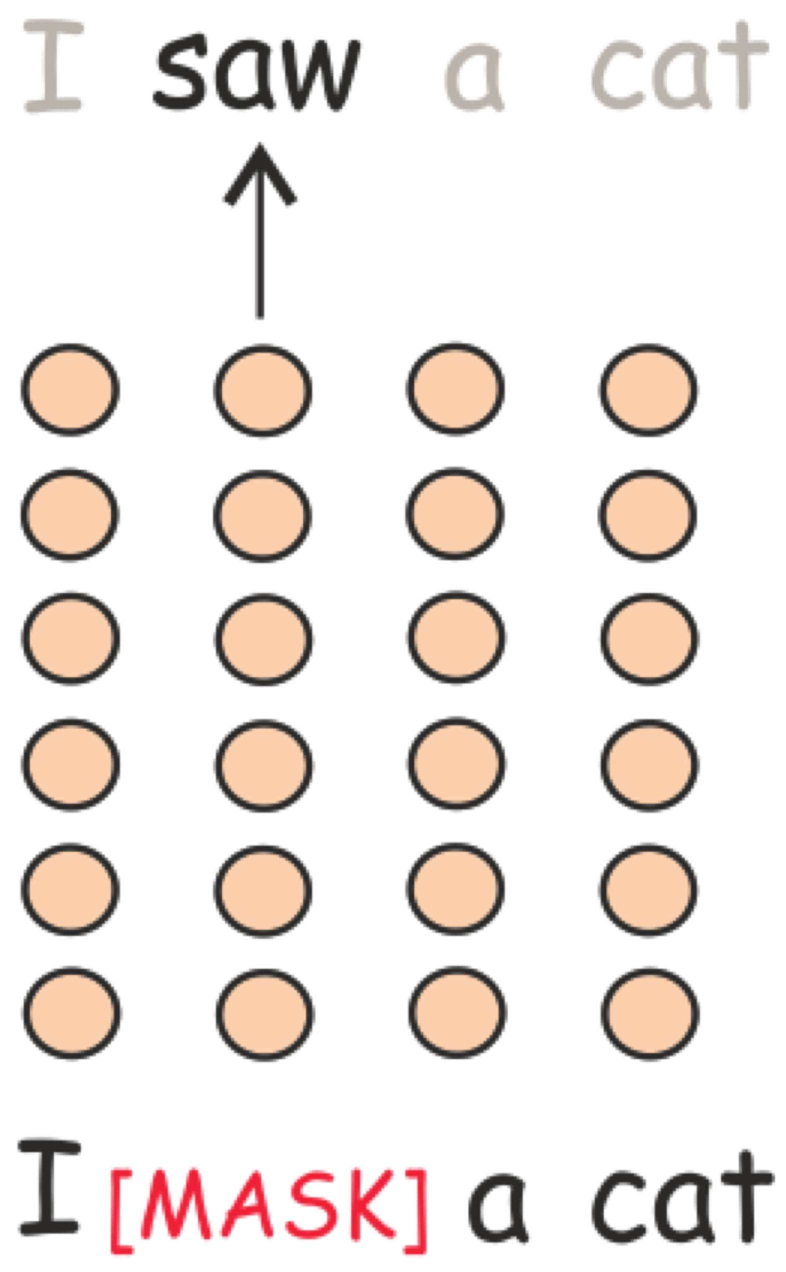
- Evolution of representations of individual tokens
- Training objectives: LM, MLM, MT
- ”Puzzles” from previous work
- The Information-Bottleneck: our point of view
- Experiments

Tasks: LM, MLM, MT

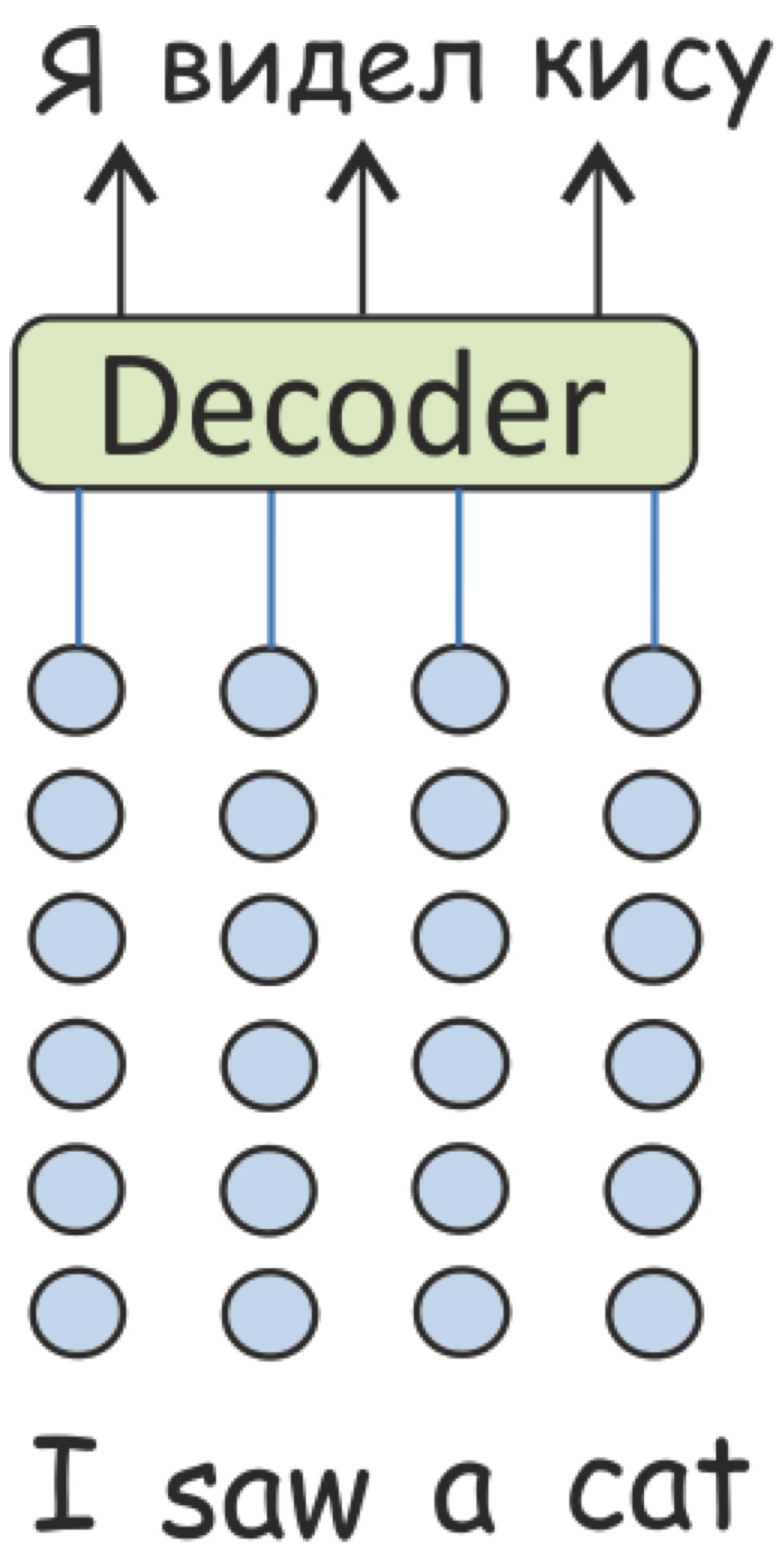
LM



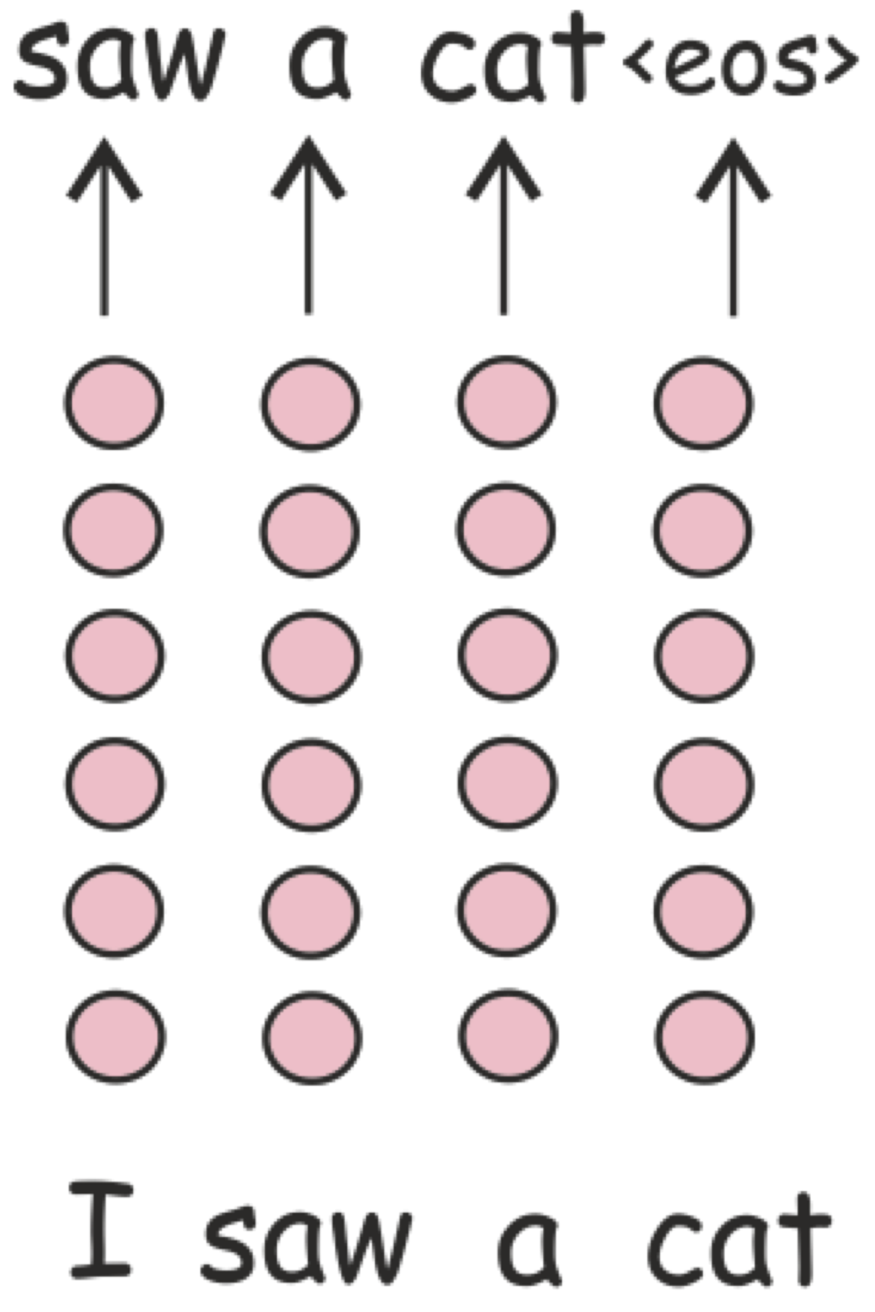
MLM



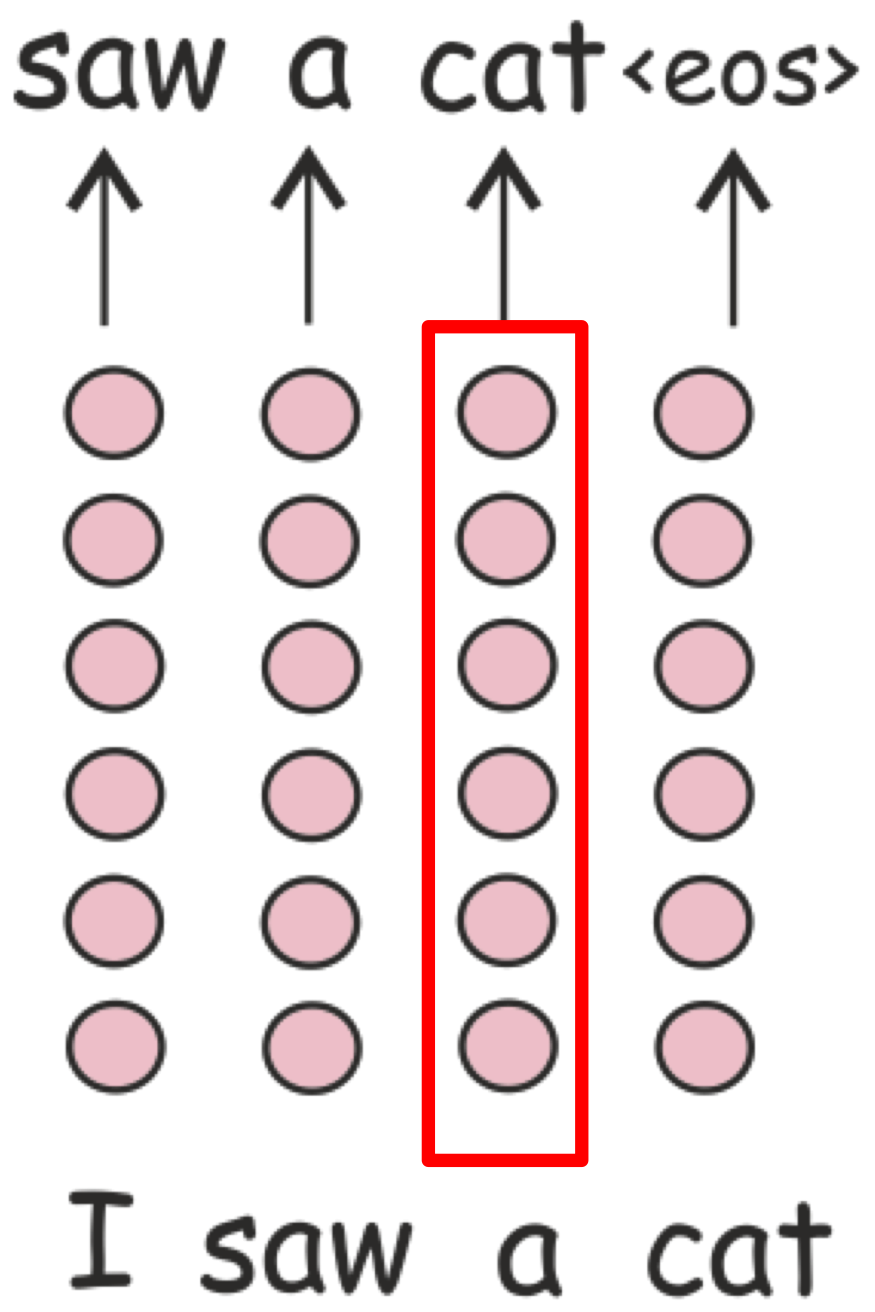
MT



LM - Language Modeling



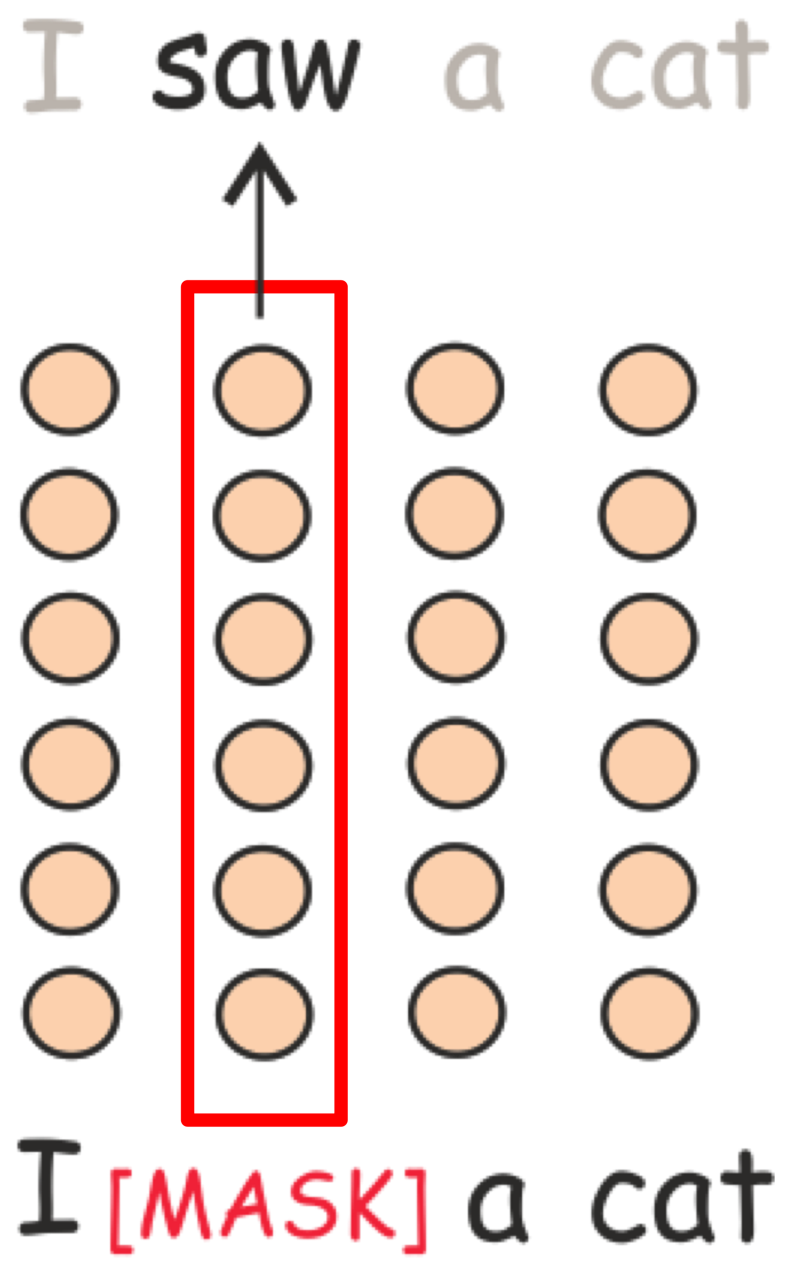
LM - Language Modeling



<u>Input</u>: current token identity and position

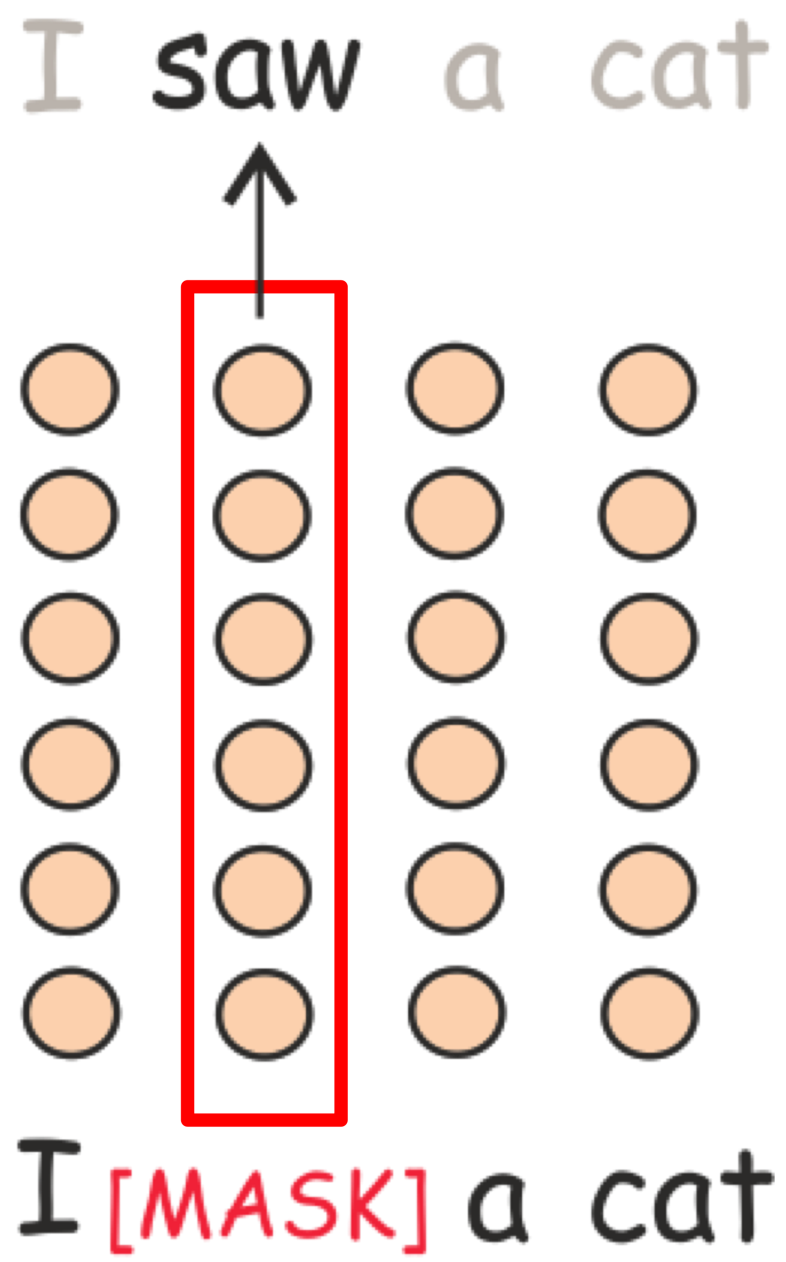
<u>Output</u>: next token

MLM – Masked Language Modeling (aka BERT)



- some tokens are selected (with probability p=15%)
- selected tokens are either replaced with [mask], random or current token

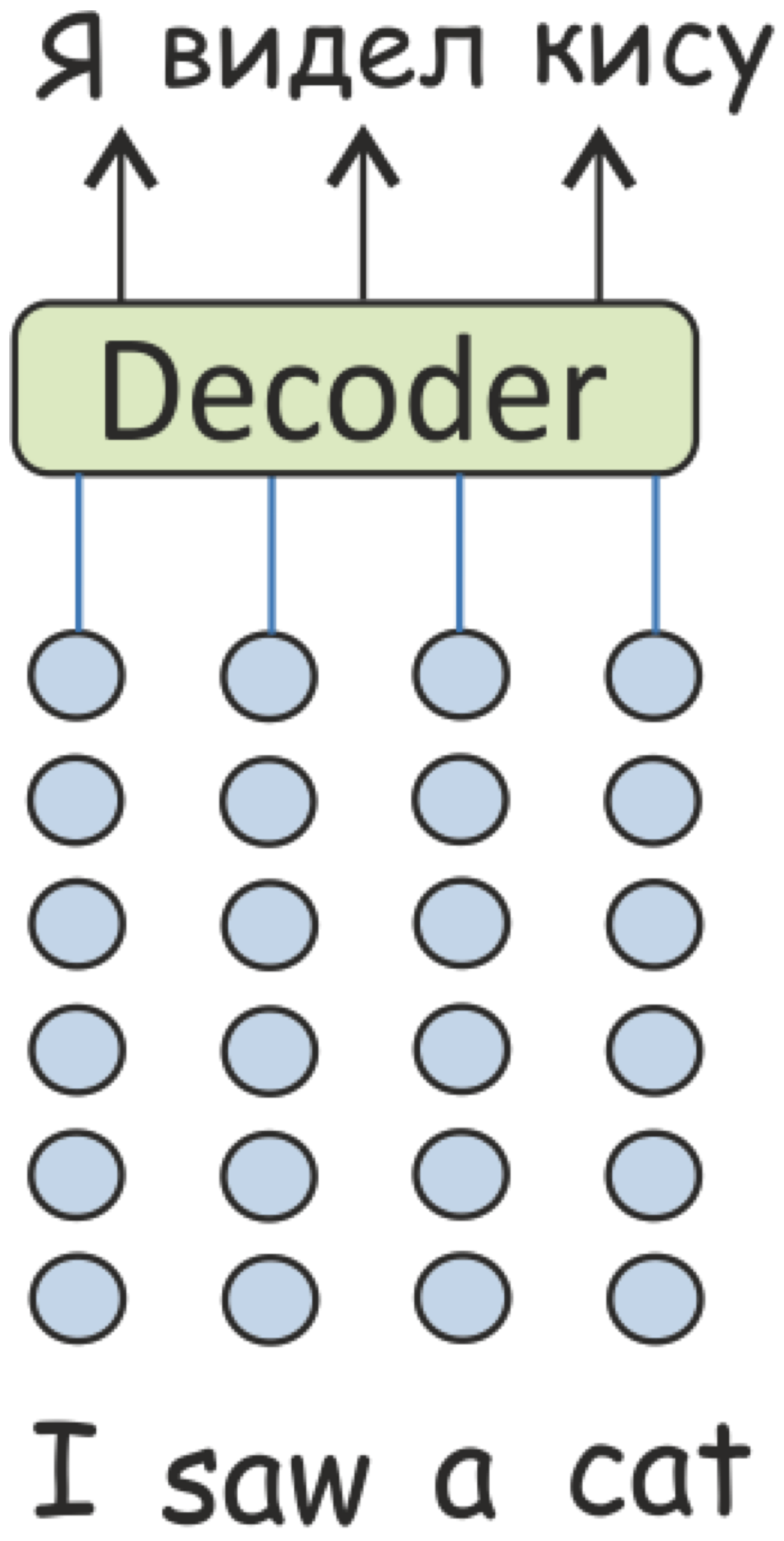
MLM – Masked Language Modeling (aka BERT)



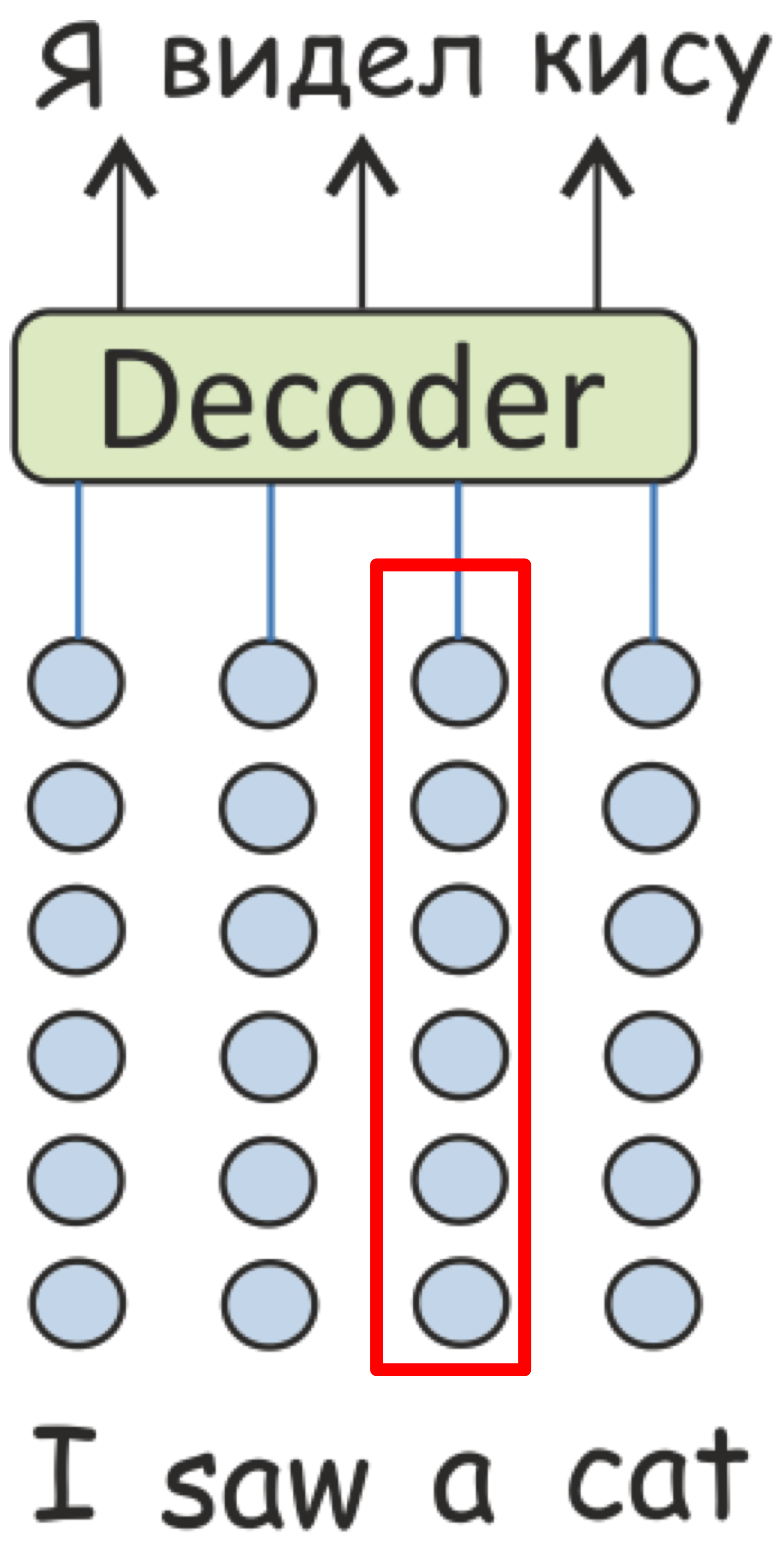
Input: [mask], random or current token identity and position

Output: current token

MT – Machine Translation



MT – Machine Translation



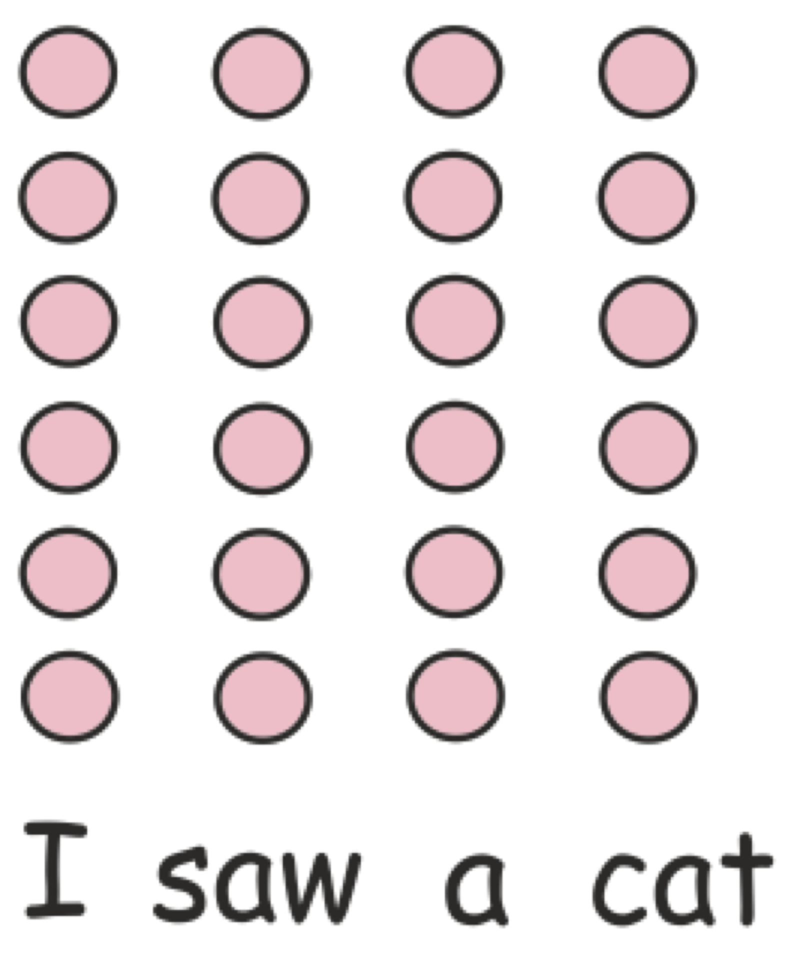
Input: current token identity and position

Output: nothing is predicted directly

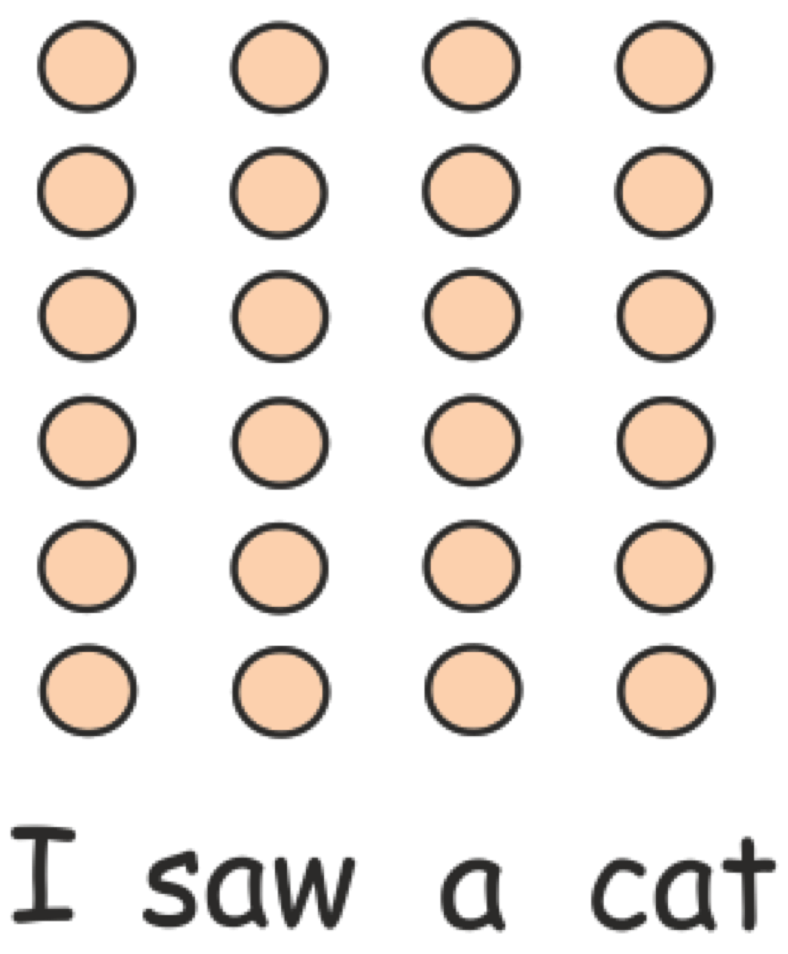
The bottom-up evolution

- Fix: model and training data
- Vary: training objective

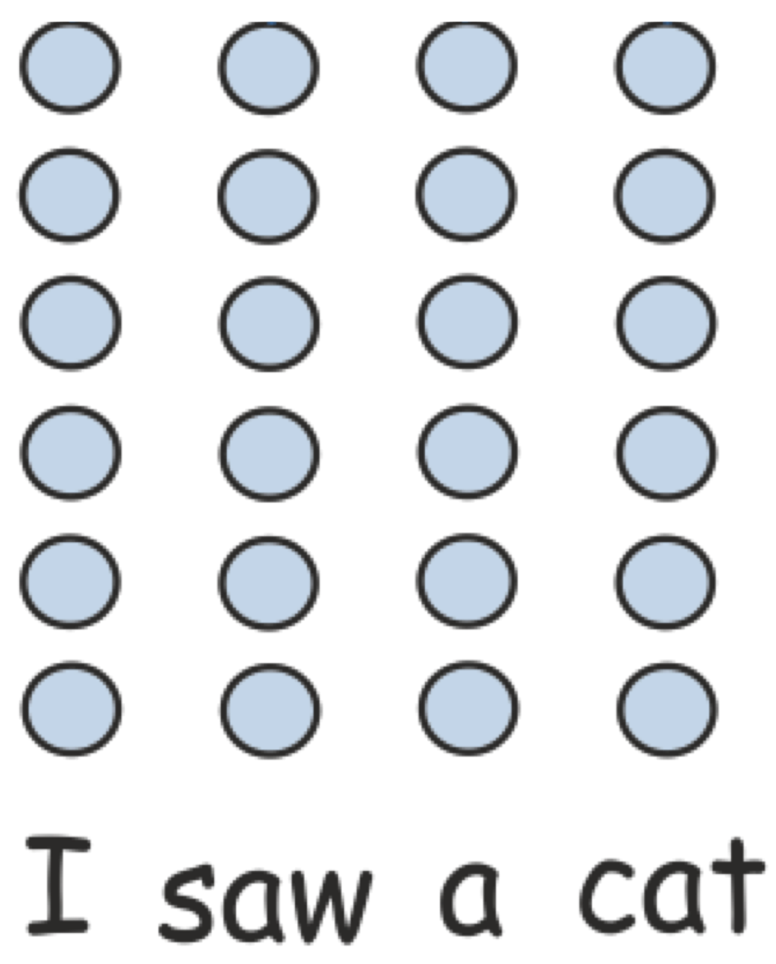
LM



MLM

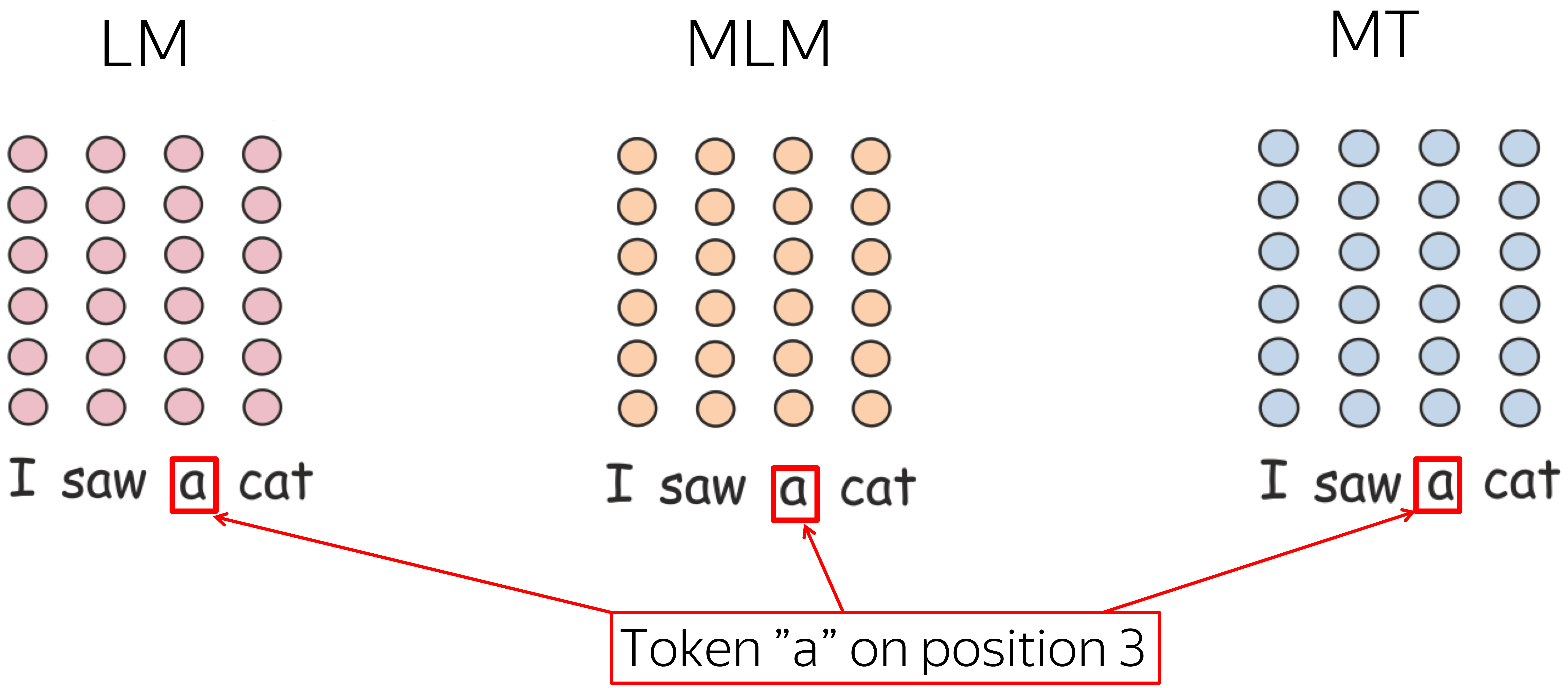


MT



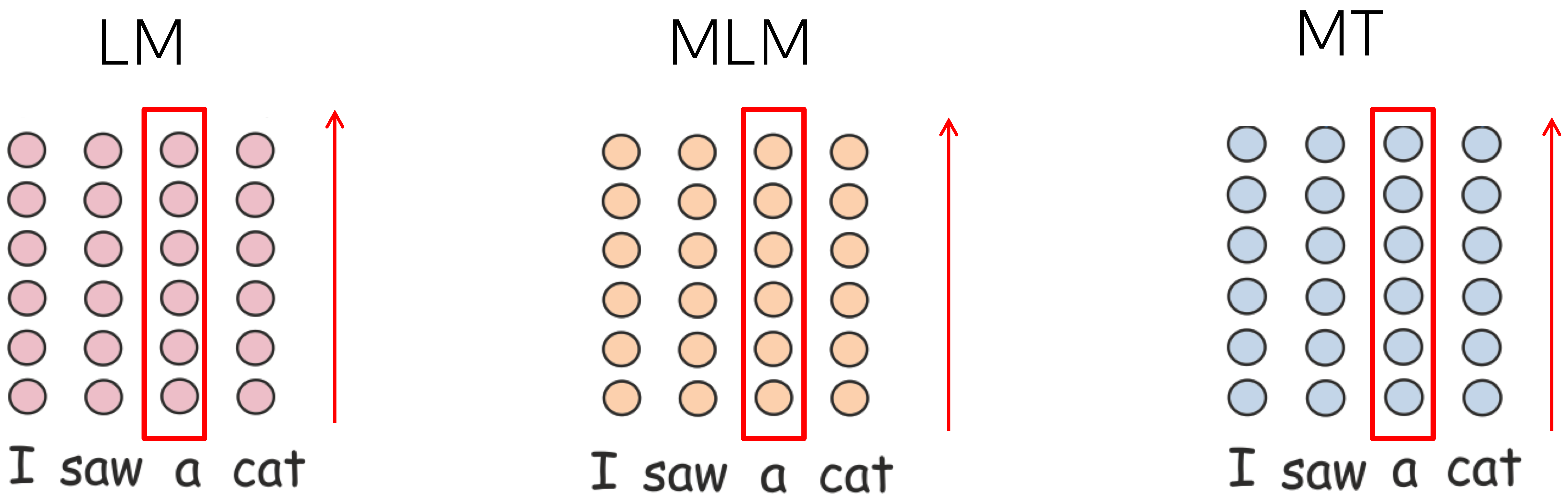
The bottom-up evolution

- Fix: model and training data
- Vary: training objective



The bottom-up evolution

- Fix: model and training data
- Vary: training objective



Plan

- Evolution of representations of individual tokens
- Training objectives: LM, MLM, MT
- ”Puzzles” from previous work
- The Information-Bottleneck: our point of view
- Experiments

Previous work: “puzzling” results

Untrained LSTMs are better for token prediction

- Untrained LSTMs outperform trained ones for word identity prediction task (Zhang & Bowman, 2018)

MT behavior is monotonic, LM is not

- For constituent labeling prediction, MT shows monotonic behavior, while LM non-monotonic (Blevins et al, 2018)

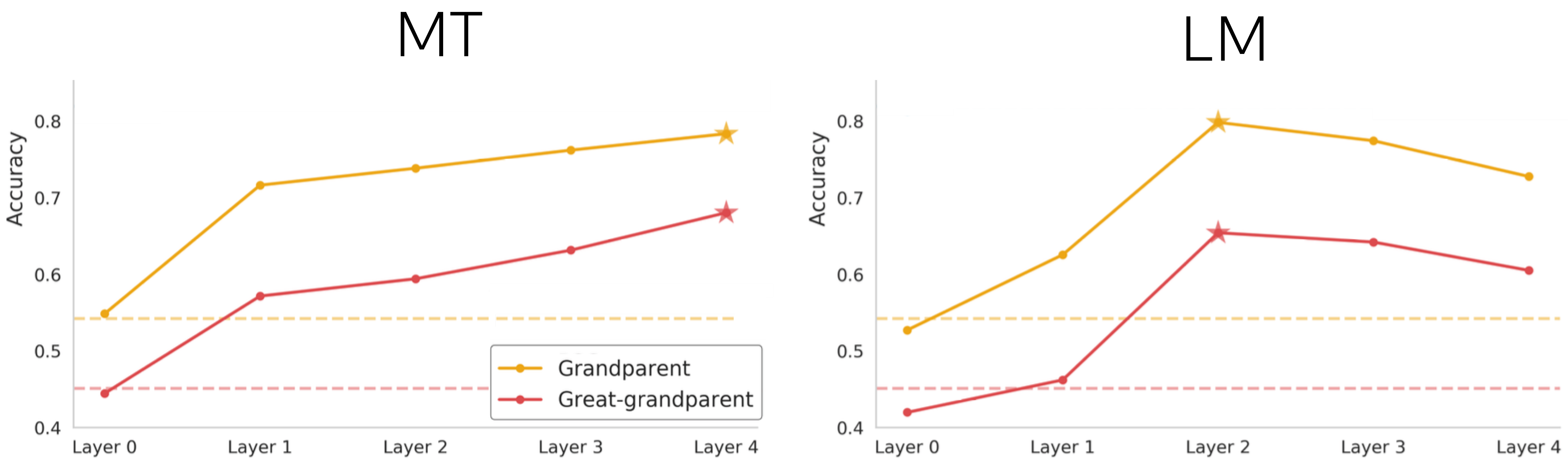


Illustration is from the original paper by Blevins et al, 2018

BERT behavior is not monotonic

- For different tasks the contribution of a layer to a task increases up to a certain layer, but then decreases at the top layers (Tenney et al, 2019)

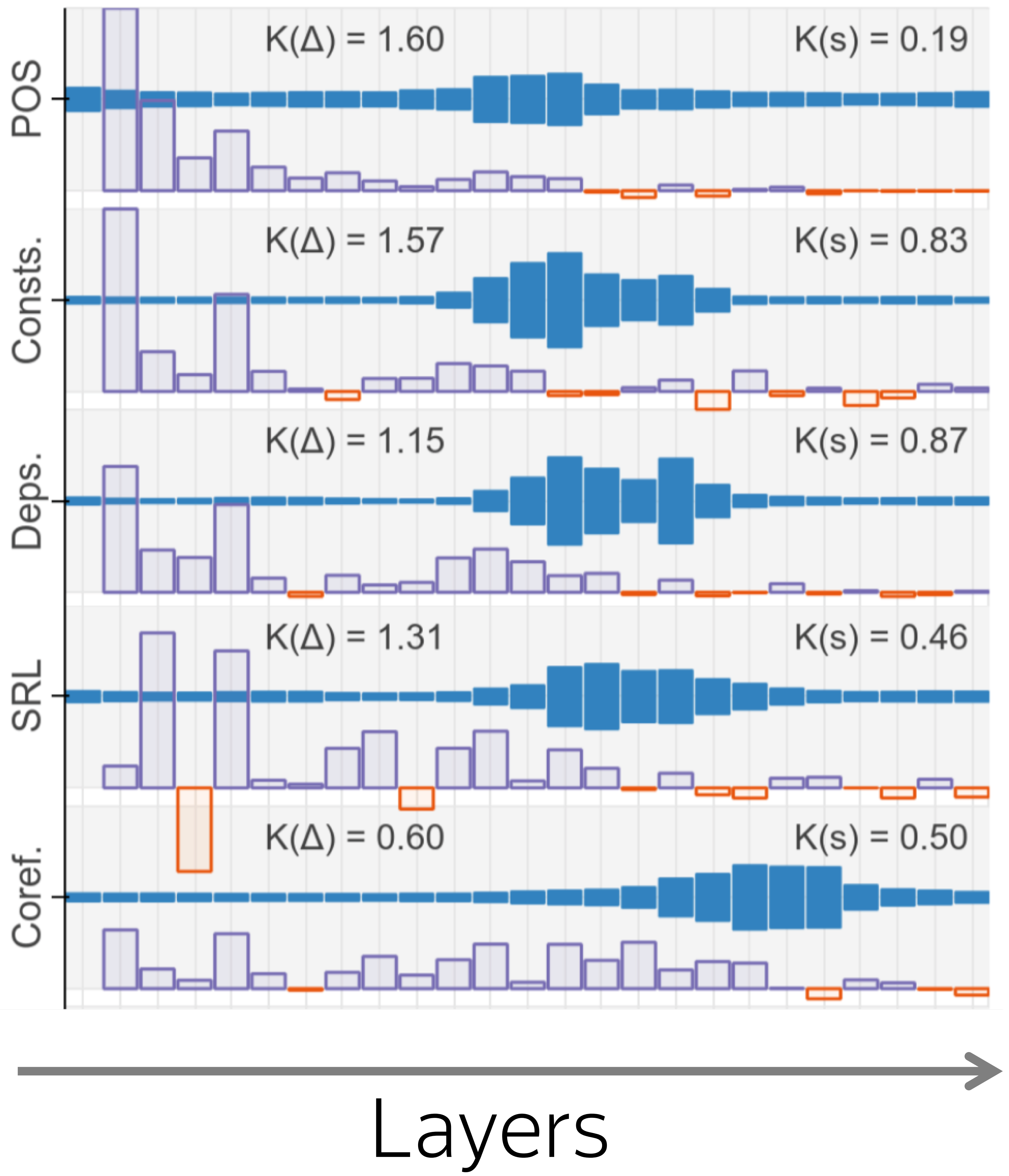
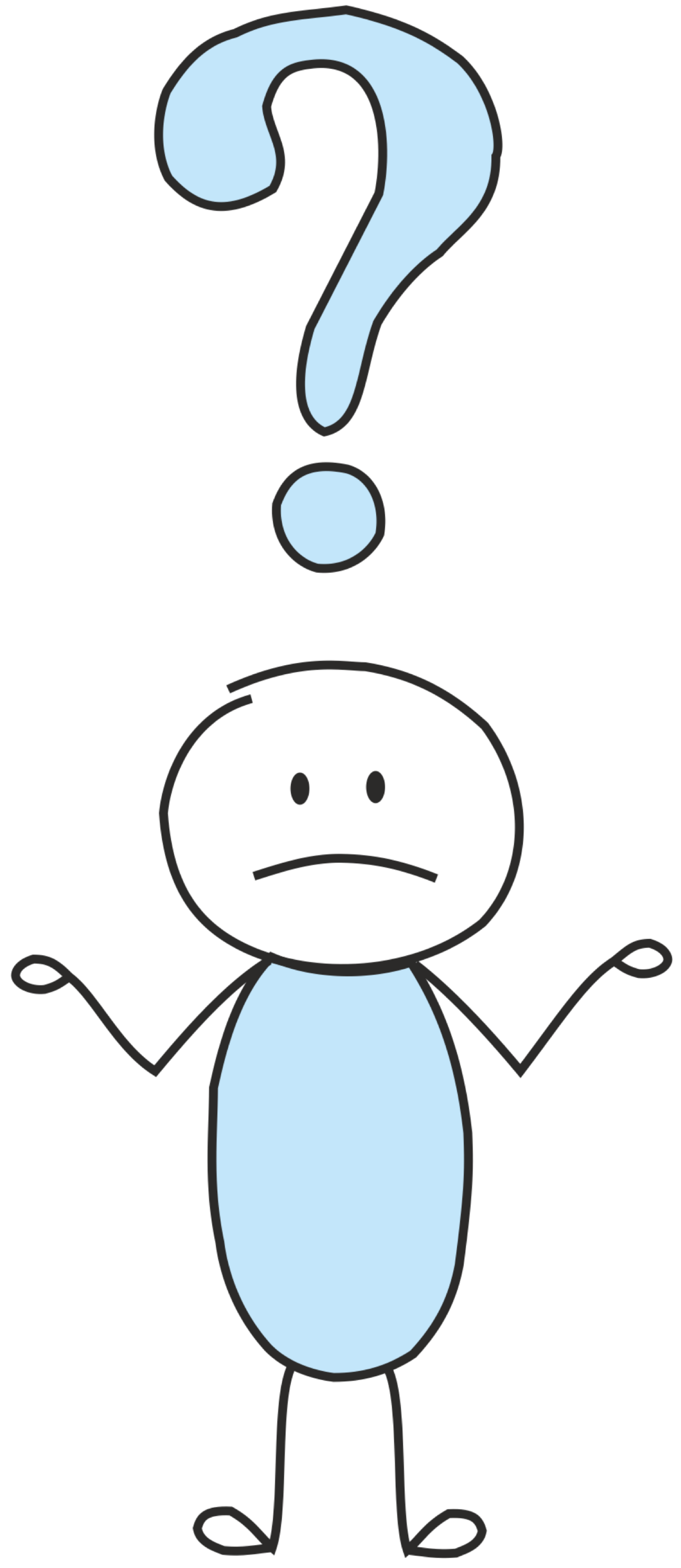


Illustration is from the original paper by Tenney et al, 2019

Why is this happening?

Problems:

- Evidence is somewhat anecdotal
- No explanation of the process behind such behavior



Plan

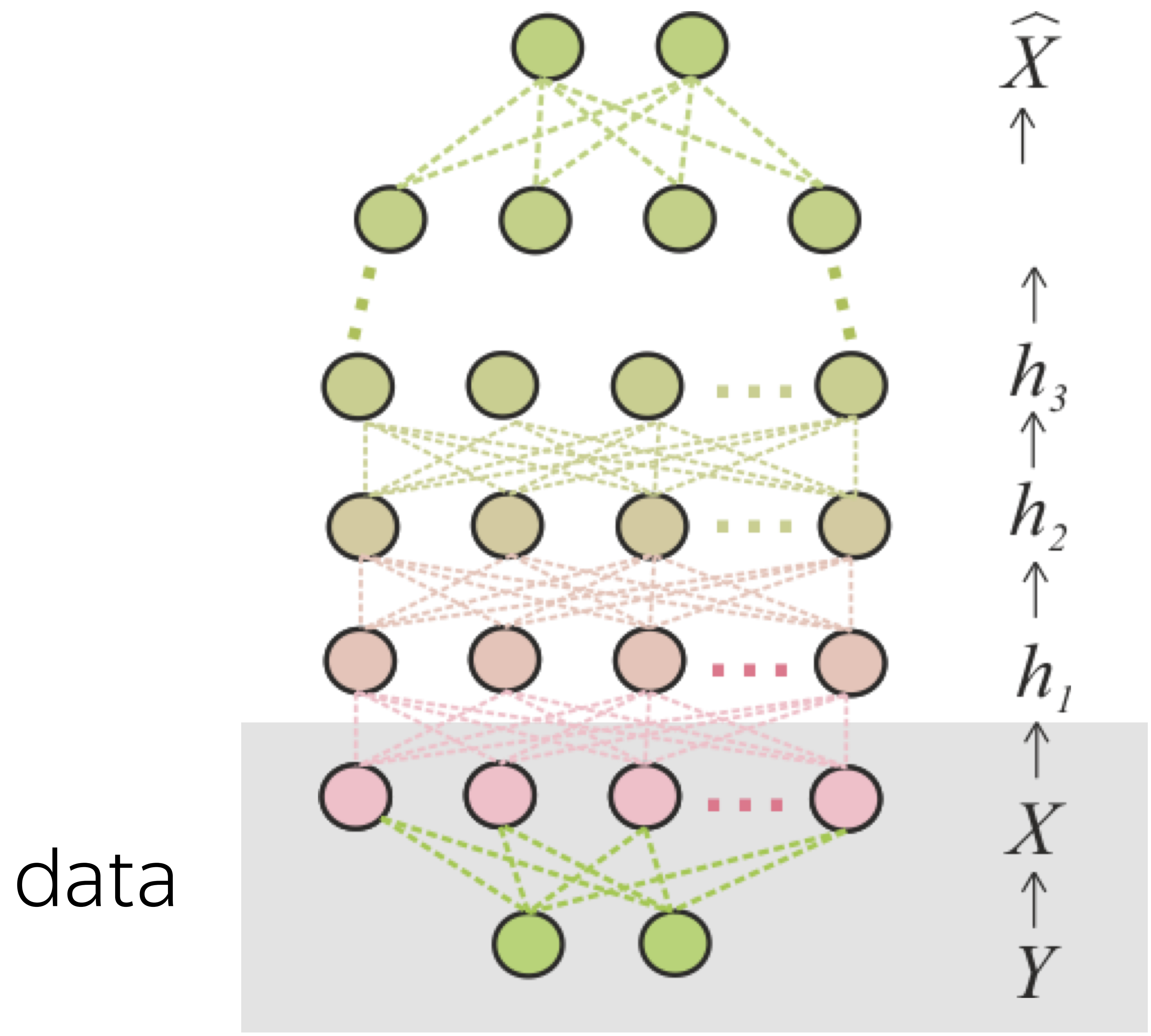
- Evolution of representations of individual tokens
- Training objectives: LM, MLM, MT
- ”Puzzles” from previous work
- The Information-Bottleneck: our point of view
- Experiments

Plan

- Evolution of representations of individual tokens
- Training objectives: LM, MLM, MT
- ”Puzzles” from previous work
- The Information-Bottleneck: our point of view
- Experiments

The Information-Bottleneck Viewpoint

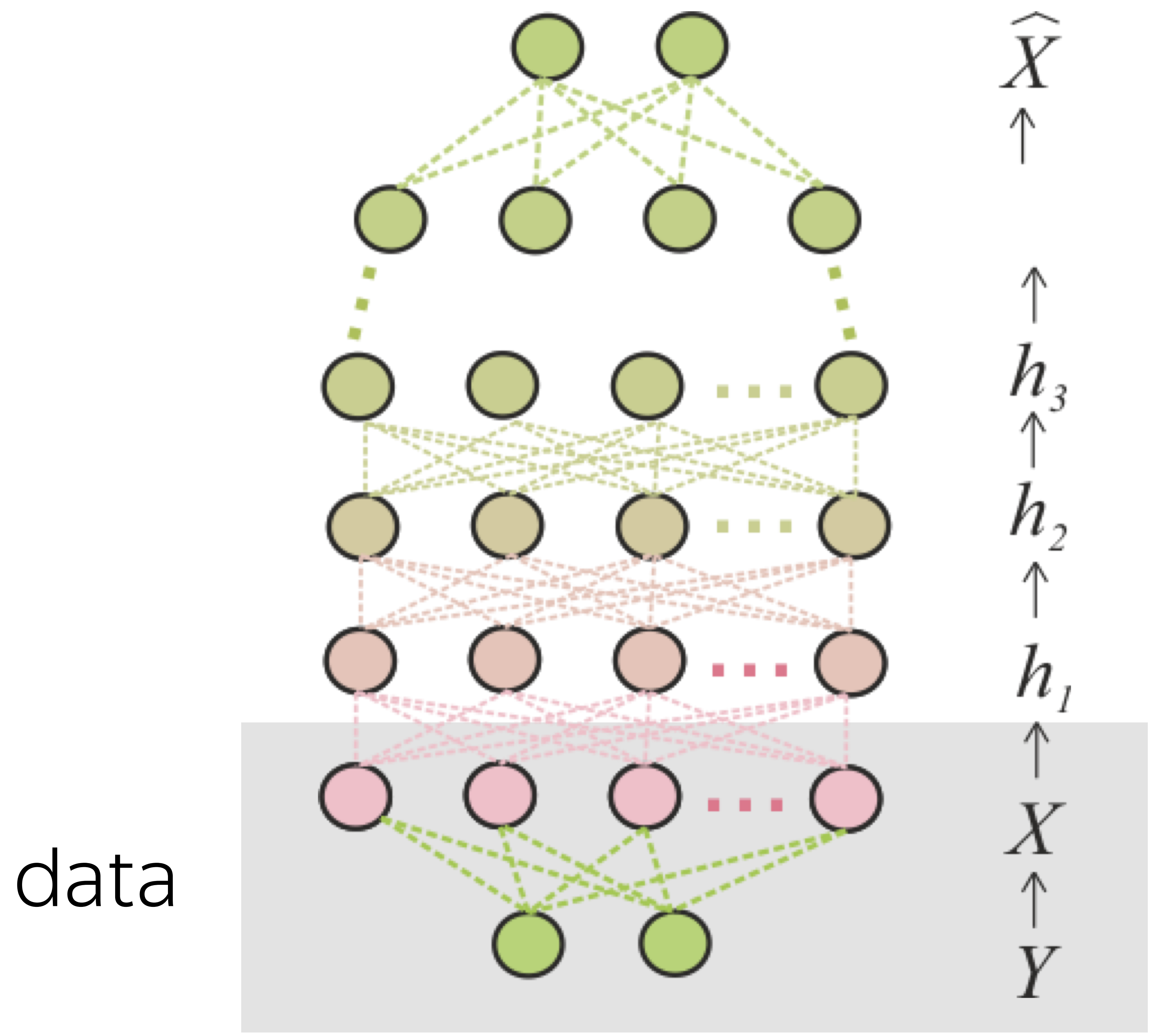
Information Bottleneck



The IB method:

$$\hat{X}: \; I(\hat{X}, X) - \beta\, I(\hat{X}, Y) \to min, \beta > 0$$

Information Bottleneck



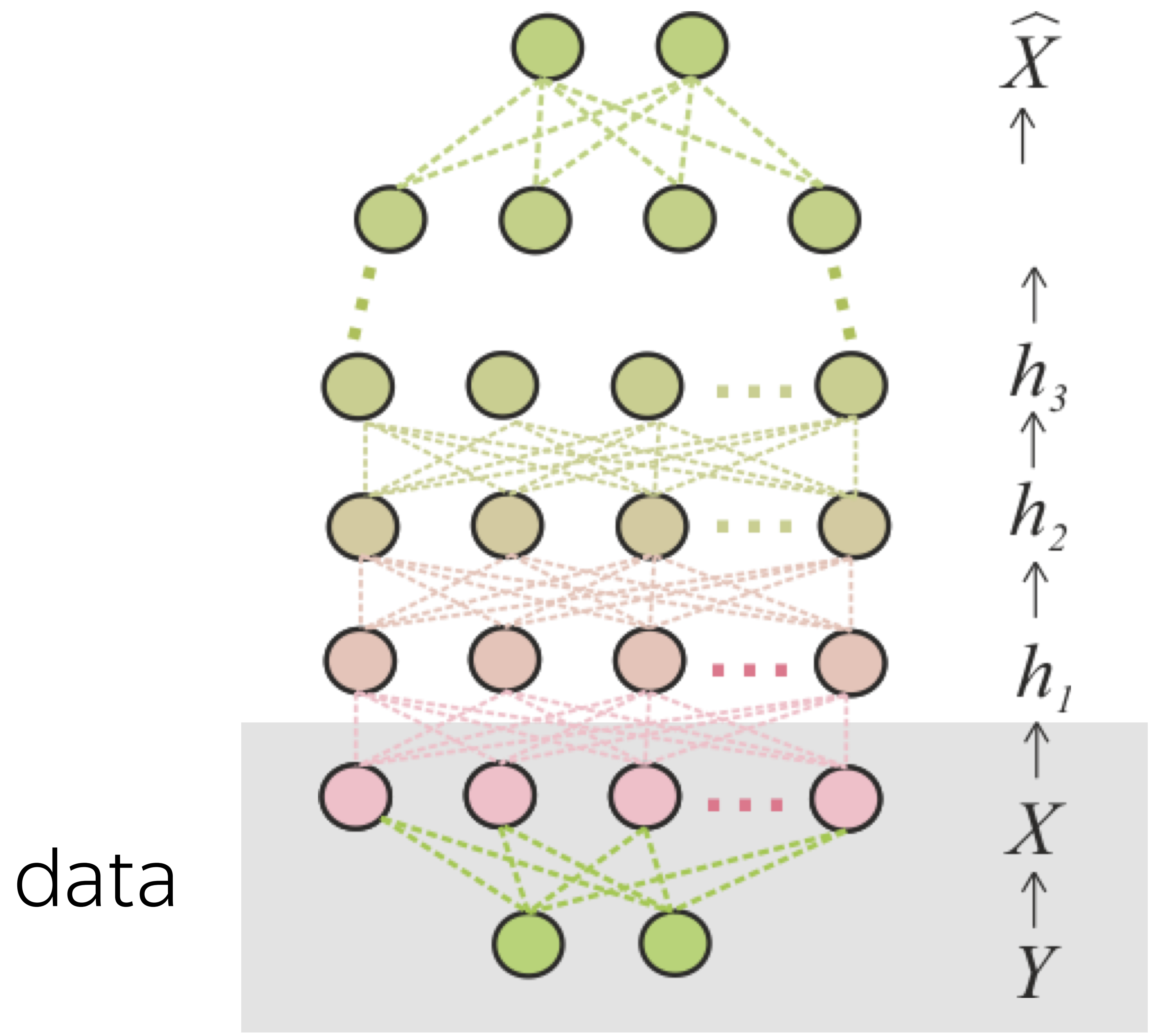
Evolution towards the theoretical optimum of the IB objective

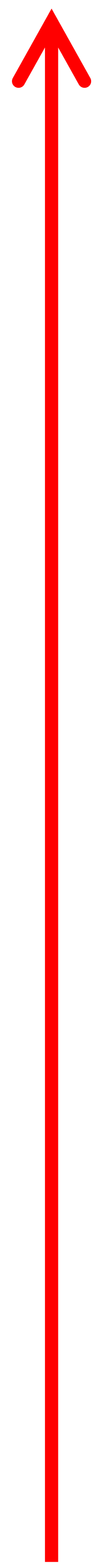
The IB method:

$$\hat{X}: \; I(\hat{X},X) - \beta\, I(\hat{X},Y) \to min, \beta > 0$$

In neural networks:

Information Bottleneck





Evolution towards the theoretical optimum of the IB objective

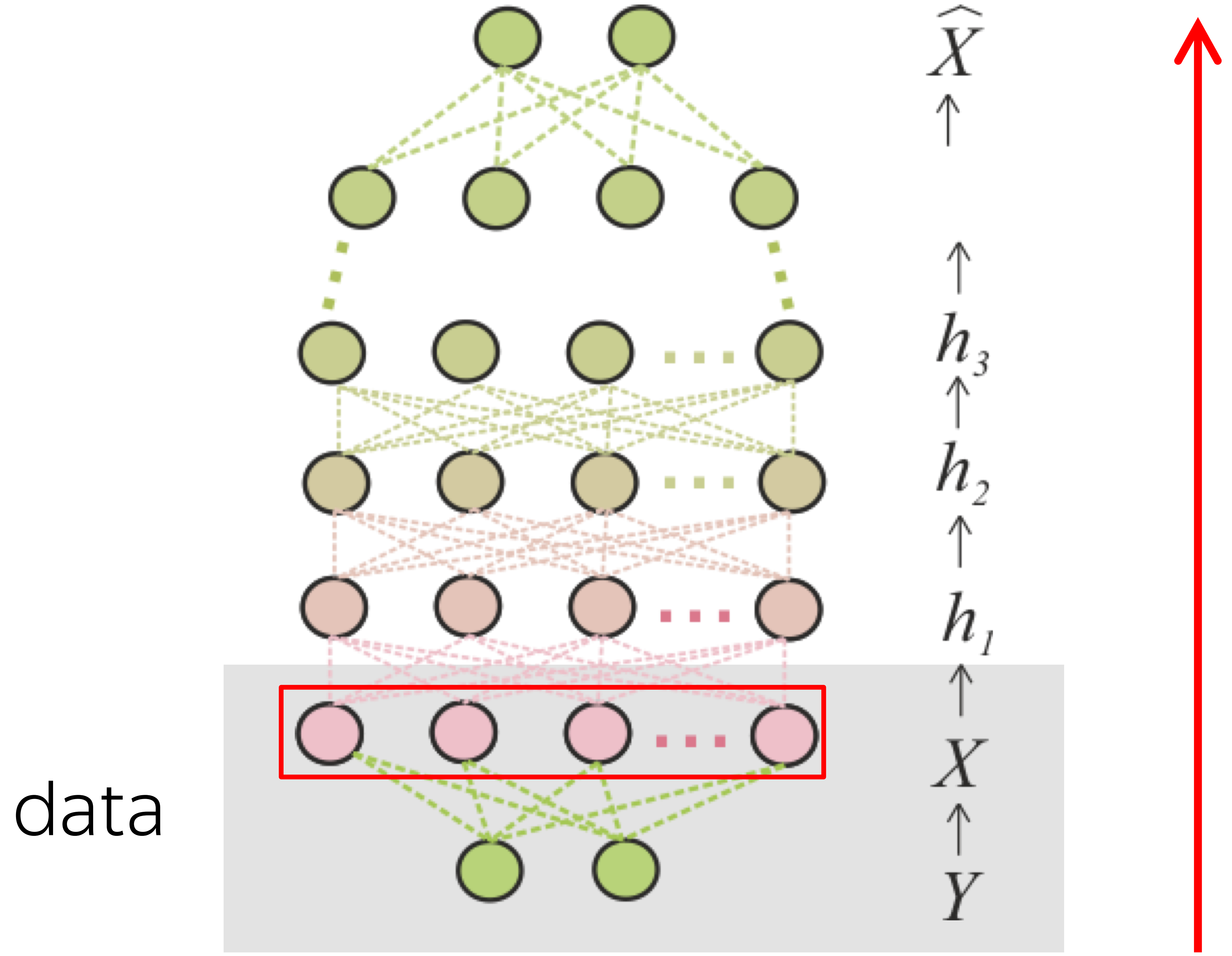
The IB method:

$$\hat{X}: \ I(\hat{X}, X) - \beta \, I(\hat{X}, Y) \to min, \beta > 0$$

In neural networks:

- a sequence of layers is a Markov chain
- squeeze irrelevant to Y information while retaining relevant

Information Bottleneck



Evolution towards the theoretical optimum of the IB objective

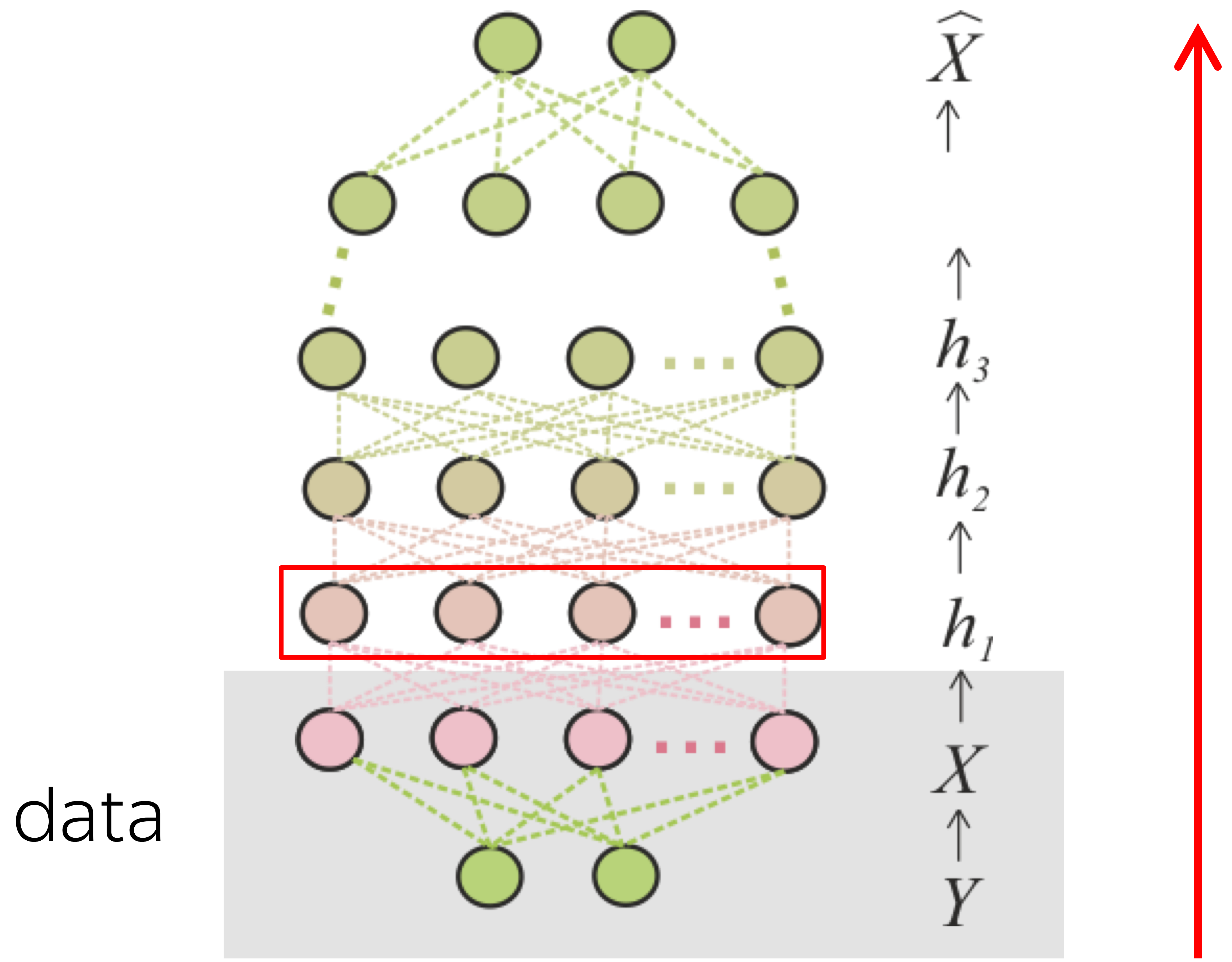
<u>The IB method</u>:

$$\hat{X}:\ I(\hat{X},X) - \beta\, I(\hat{X},Y) \rightarrow min, \beta > 0$$

<u>In neural networks</u>:

- a sequence of layers is a Markov chain
- squeeze irrelevant to Y information while retaining relevant

Information Bottleneck



Evolution towards the theoretical optimum of the IB objective

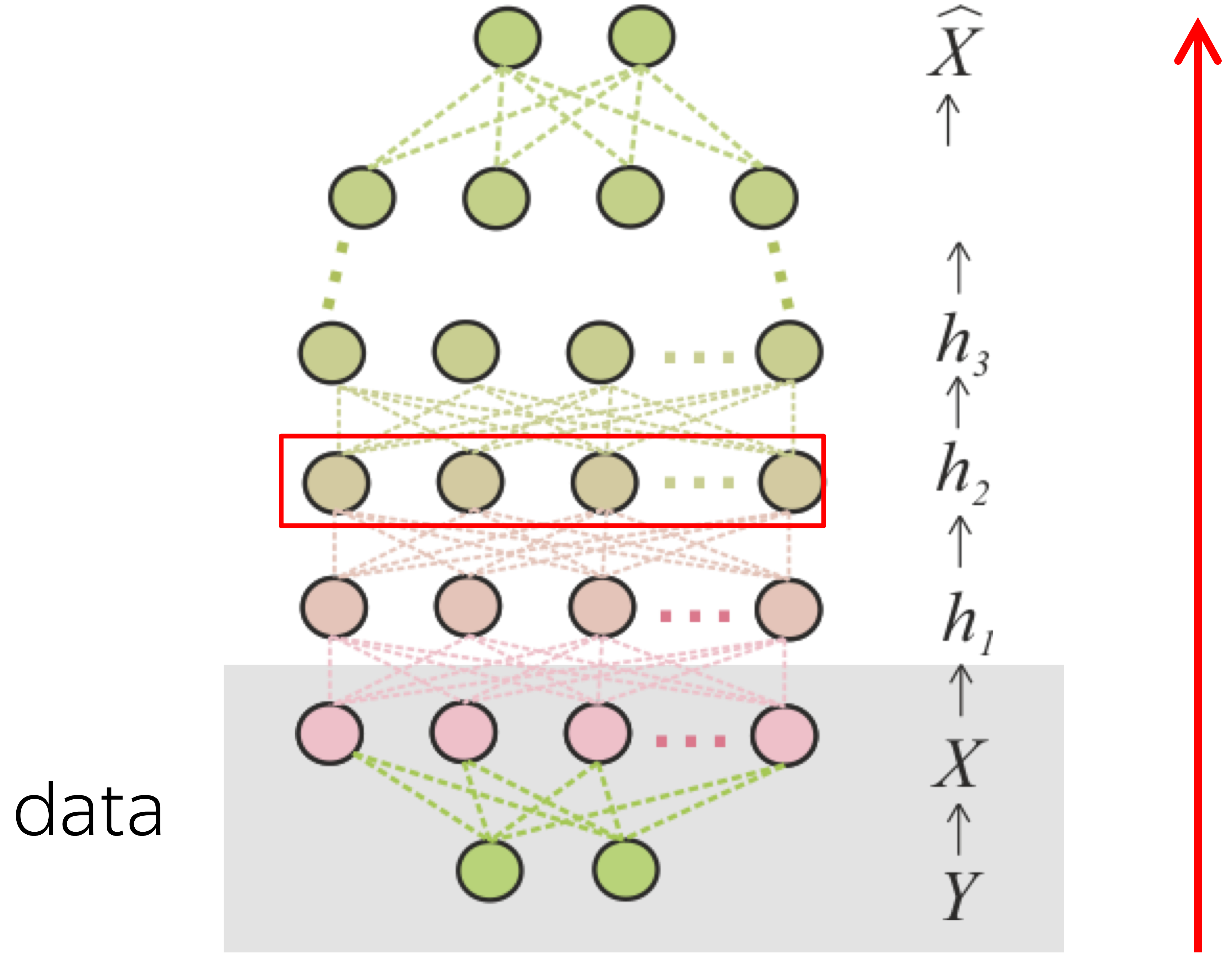
The IB method:

$$\hat{X}: \; I(\hat{X}, X) - \beta\, I(\hat{X}, Y) \to min, \beta > 0$$

In neural networks:

- a sequence of layers is a Markov chain
- squeeze irrelevant to Y information while retaining relevant

Information Bottleneck



Evolution towards the theoretical optimum of the IB objective

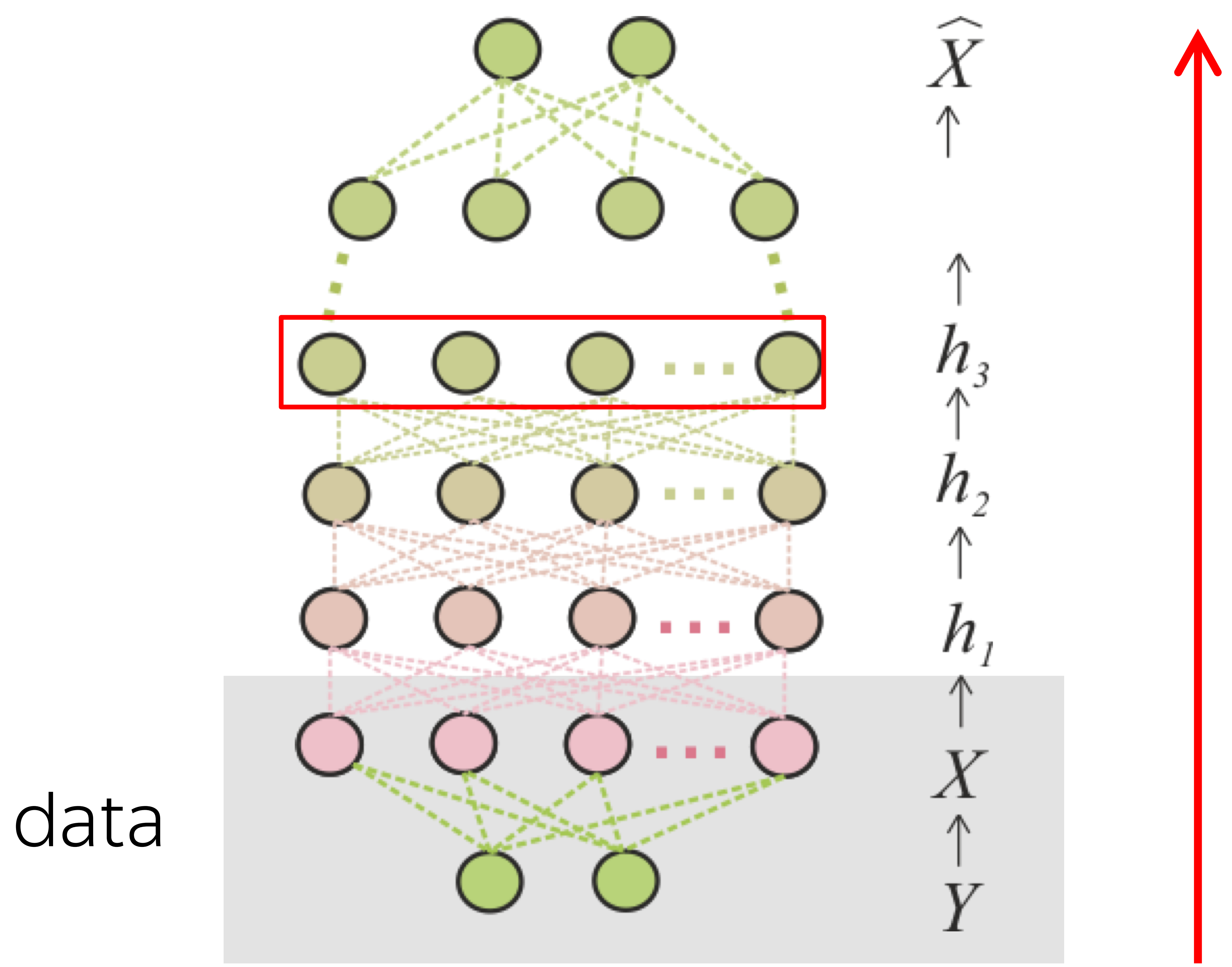
The IB method:

$$\hat{X}: \ I(\hat{X}, X) - \beta\, I(\hat{X}, Y) \to min, \beta > 0$$

In neural networks:

- a sequence of layers is a Markov chain
- squeeze irrelevant to Y information while retaining relevant

Information Bottleneck



Evolution towards the theoretical optimum of the IB objective

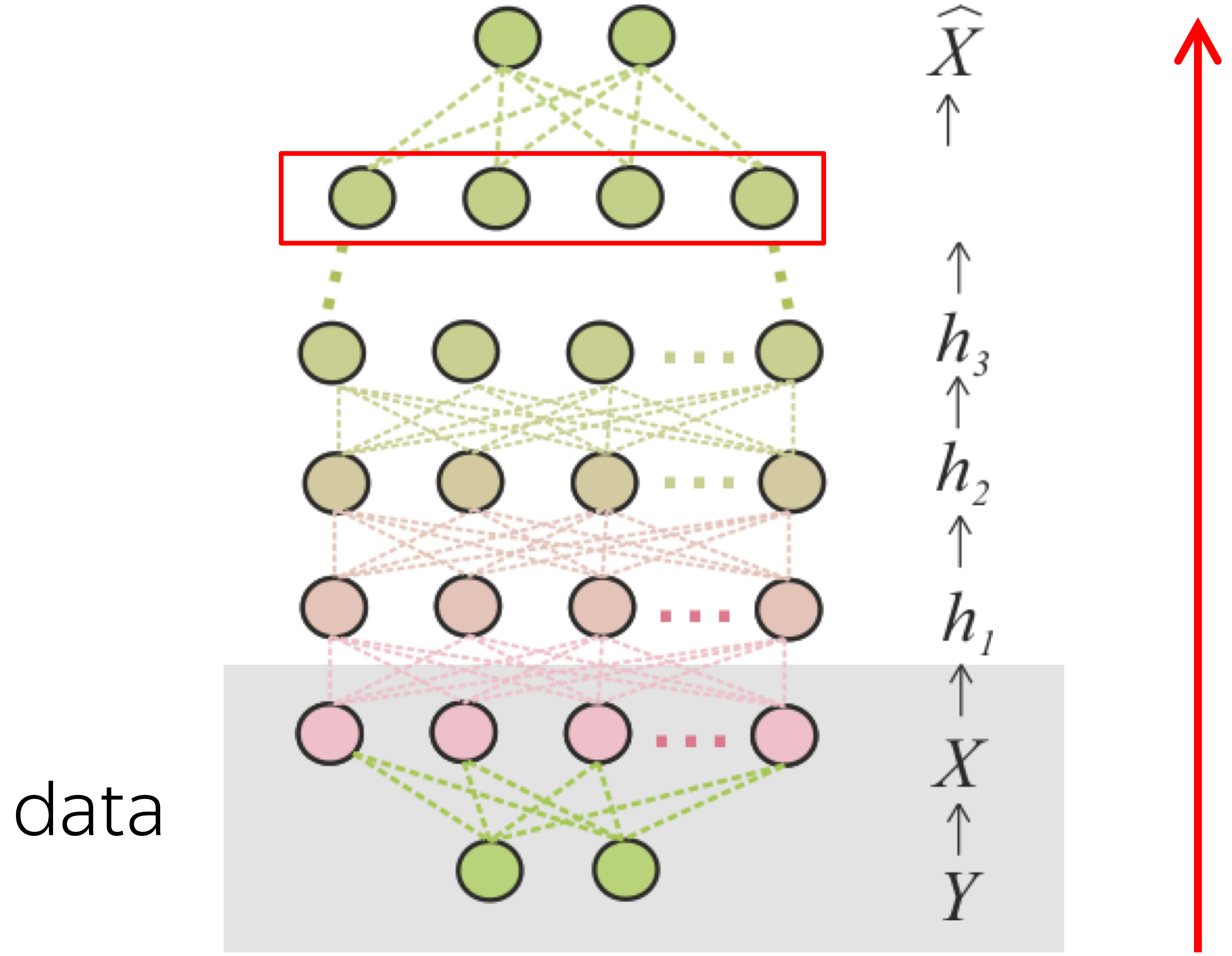
The IB method:

$$\hat{X}: \; I(\hat{X}, X) - \beta\, I(\hat{X}, Y) \rightarrow min, \beta > 0$$

In neural networks:

- a sequence of layers is a Markov chain
- squeeze irrelevant to Y information while retaining relevant

Information Bottleneck



Evolution towards the theoretical optimum of the IB objective

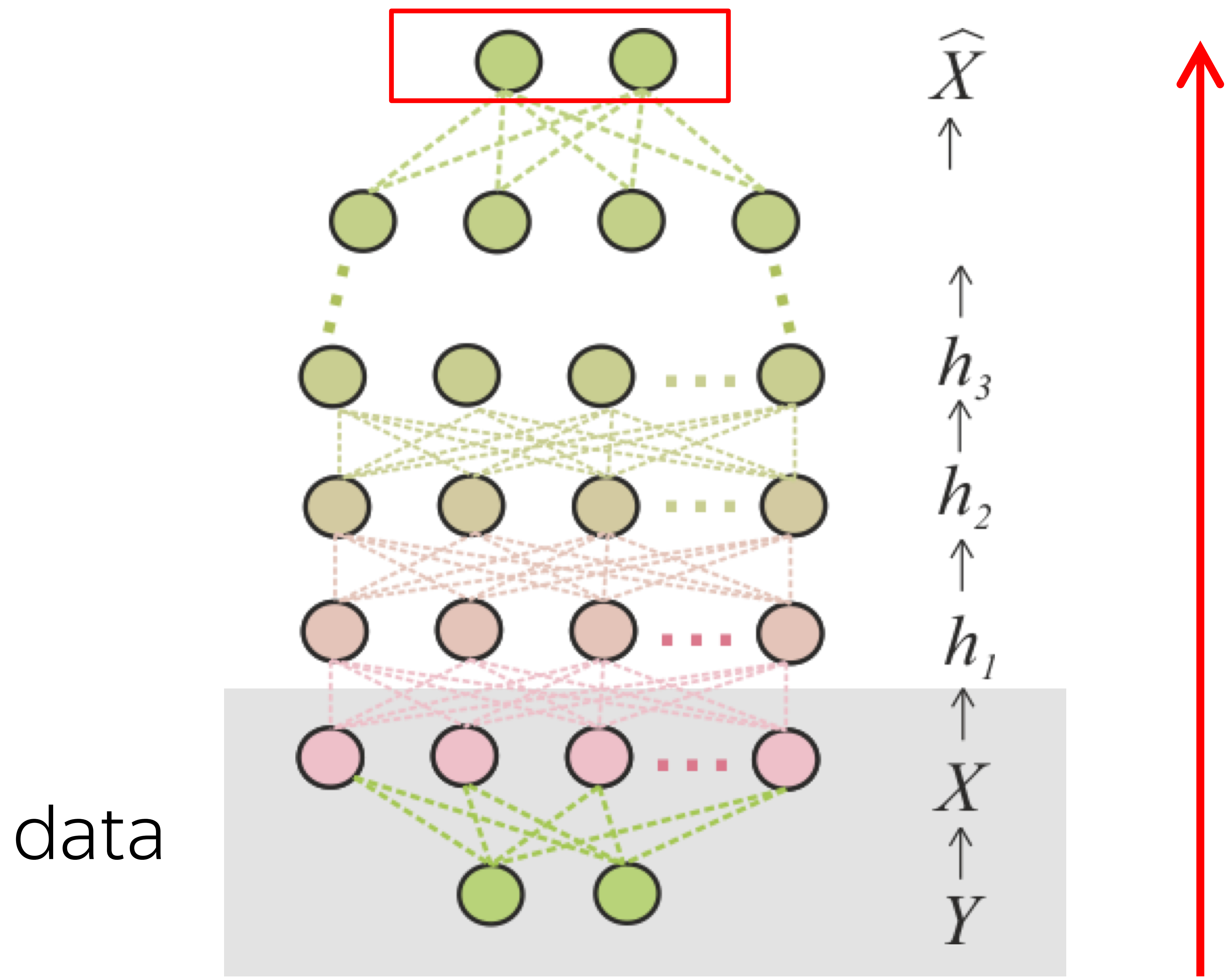
The IB method:

$$\hat{X}: \; I(\hat{X}, X) - \beta\, I(\hat{X}, Y) \rightarrow min, \beta > 0$$

In neural networks:

- a sequence of layers is a Markov chain
- squeeze irrelevant to Y information while retaining relevant

Information Bottleneck



Evolution towards the theoretical optimum of the IB objective

The IB method:

$$\hat{X}: \; I(\hat{X}, X) - \beta\, I(\hat{X}, Y) \rightarrow min, \beta > 0$$

In neural networks:

- a sequence of layers is a Markov chain
- squeeze irrelevant to Y information while retaining relevant

Plan

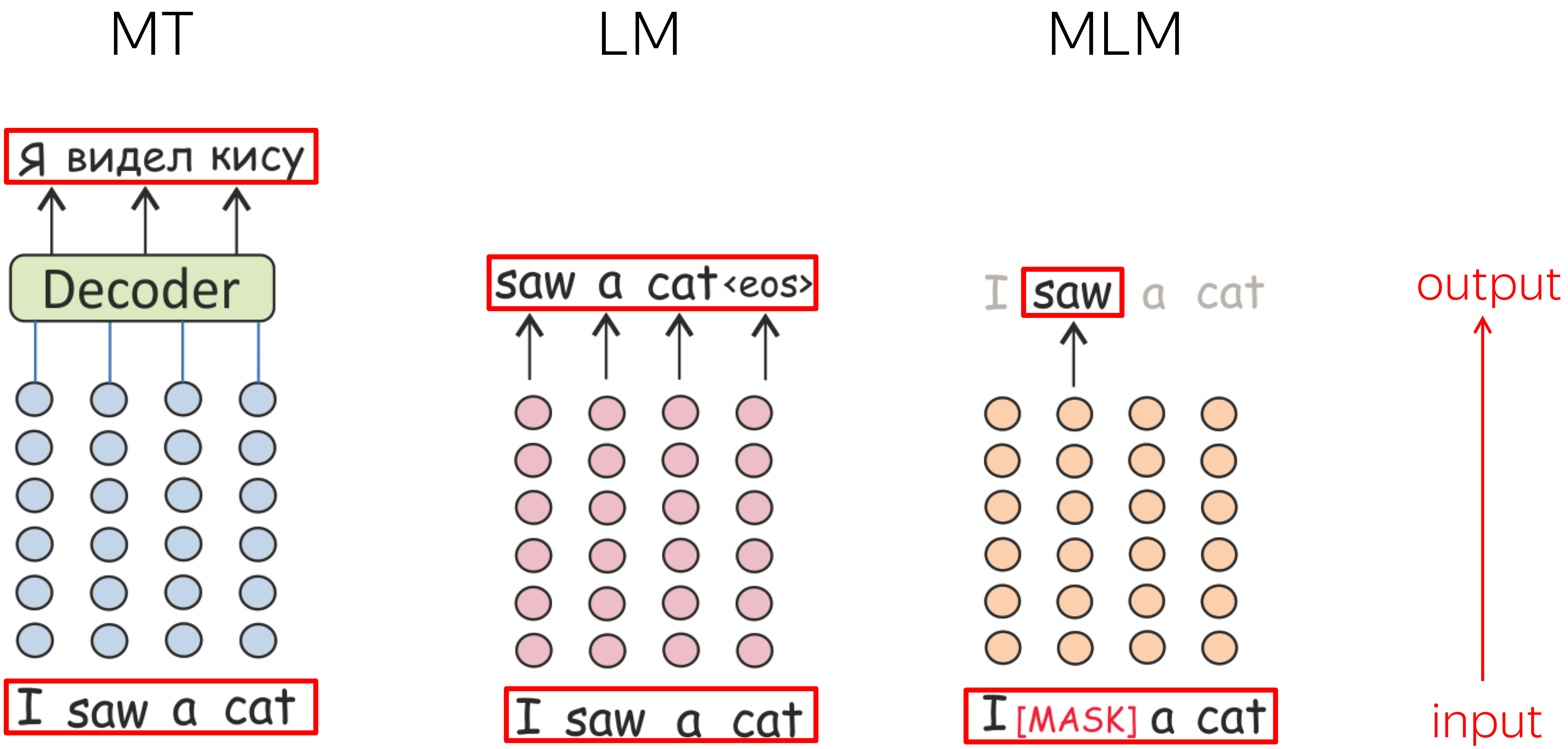
- Evolution of representations of individual tokens
- Training objectives: LM, MLM, MT
- ”Puzzles” from previous work
- The Information-Bottleneck: our point of view
- Experiments

Plan

- Evolution of representations of individual tokens
- Training objectives: LM, MLM, MT
- ”Puzzles” from previous work
- The Information-Bottleneck: our point of view
- Experiments
  - Information Bottleneck for token representations
  - …

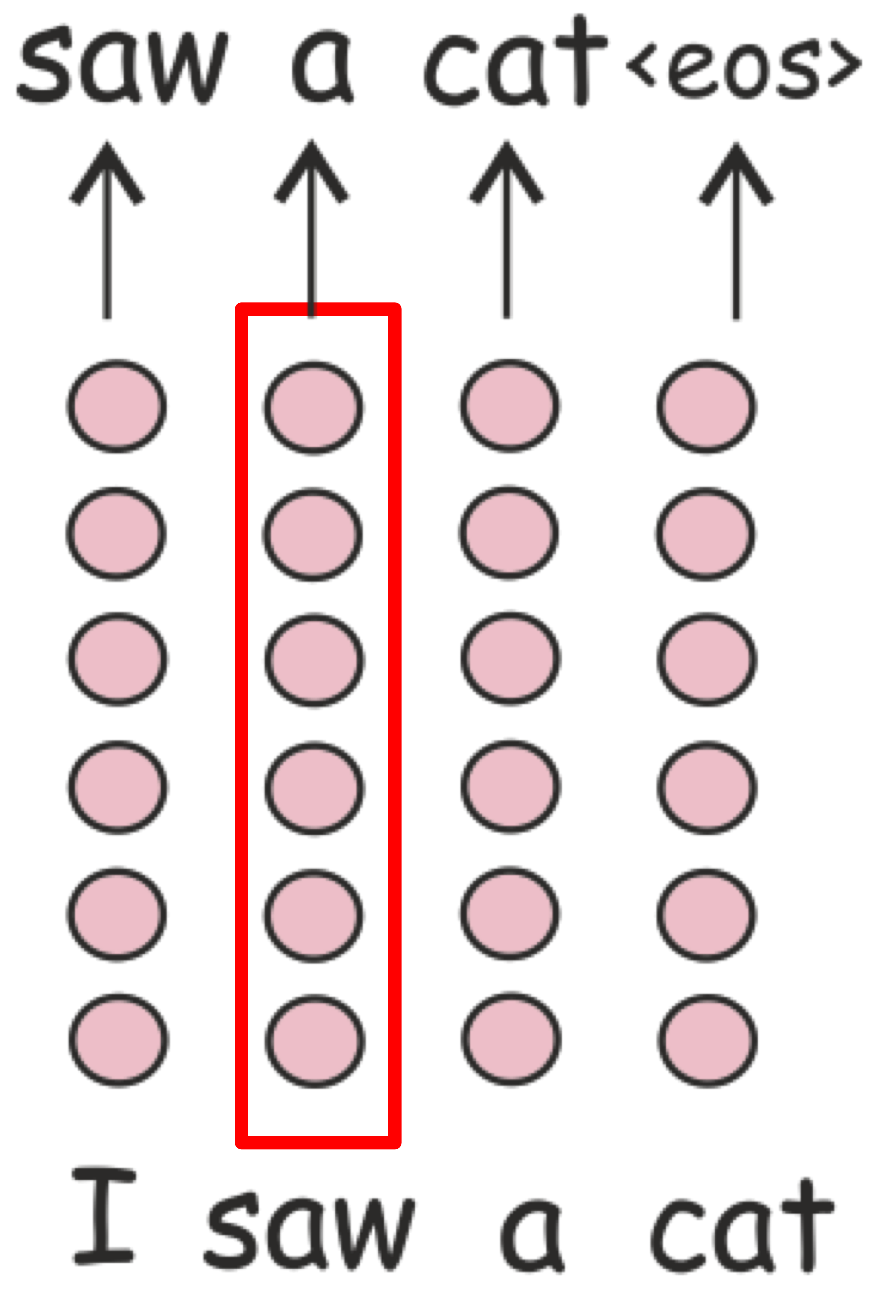
Information Bottleneck for Token Representations

Model as a function from input to output



Our setting:
representations of individual tokens

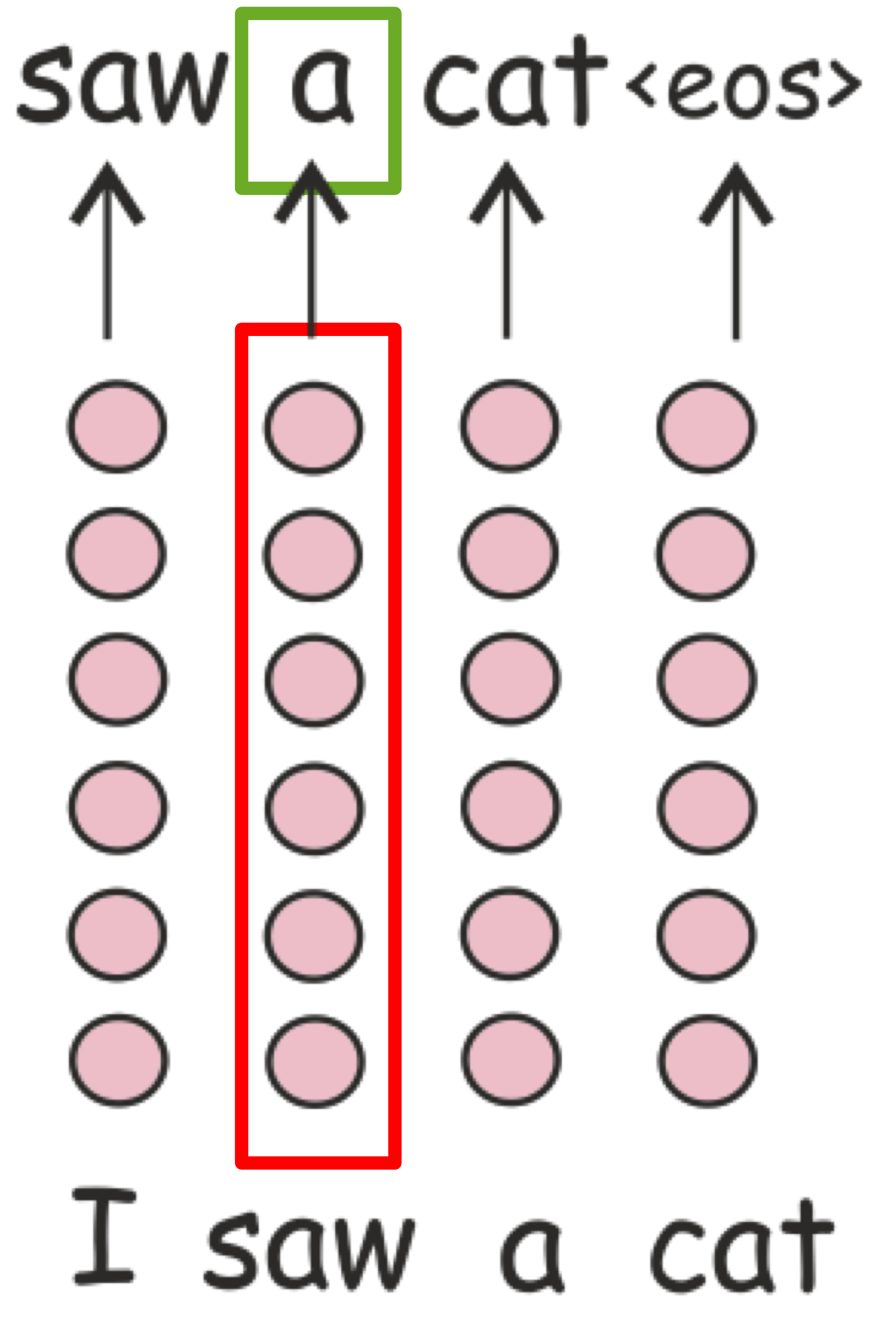
Two roles a token representation plays:



Our setting: representations of individual tokens

Two roles a token representation plays:

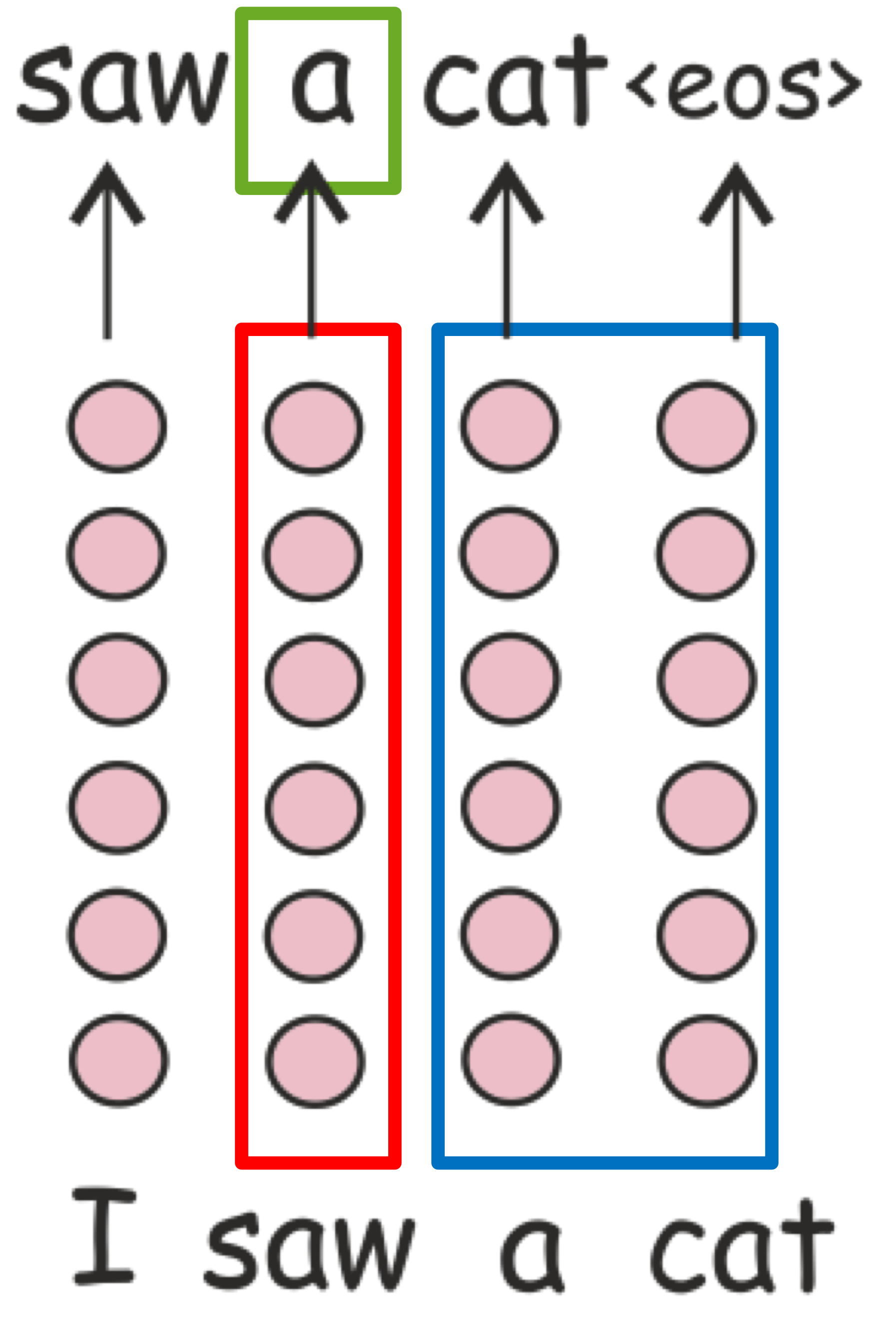
- Predicting the output label



Our setting:
representations of individual tokens

Two roles a token representation plays:

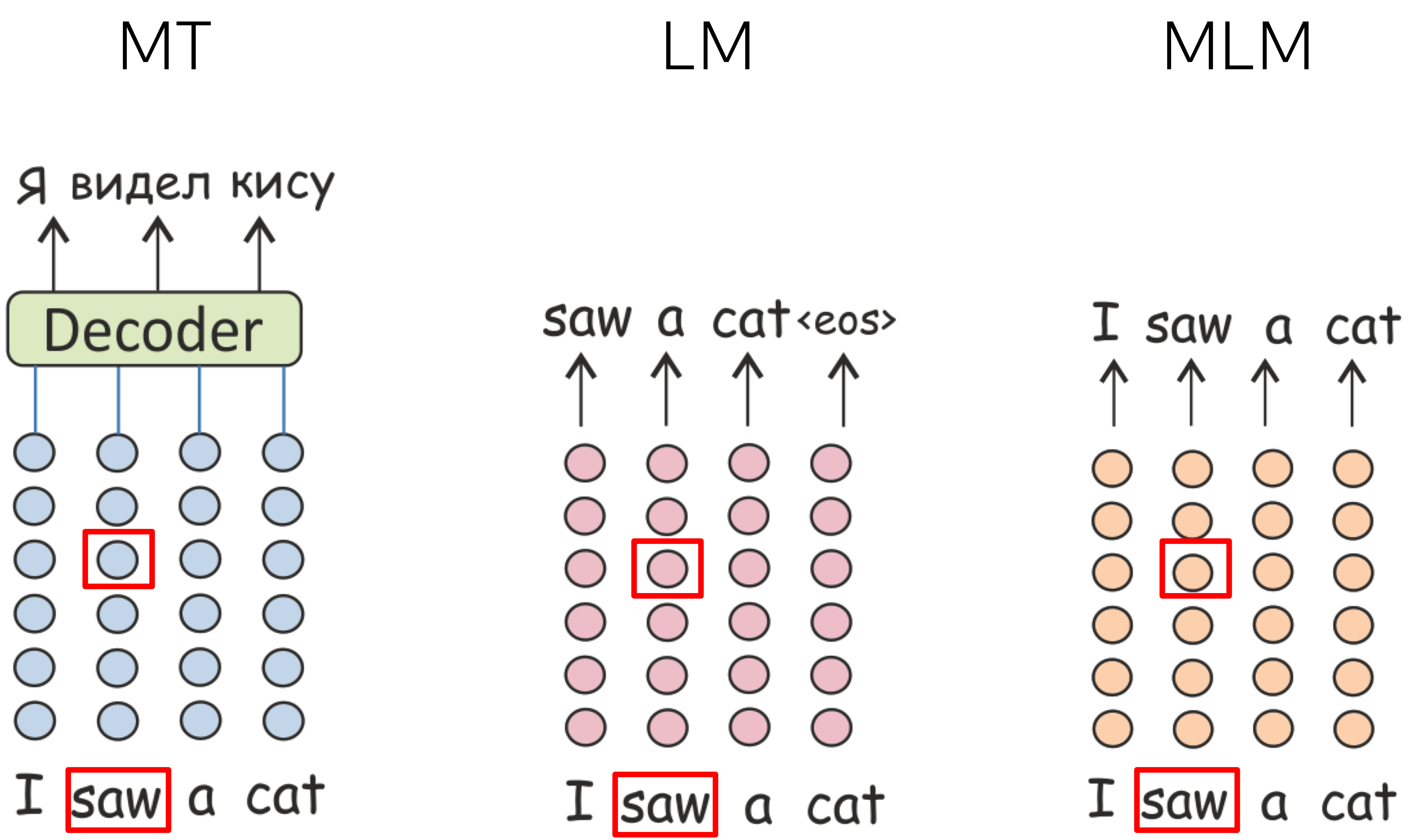
- Predicting the output label
- Preserving information necessary to build representations of other tokens



The task defines:

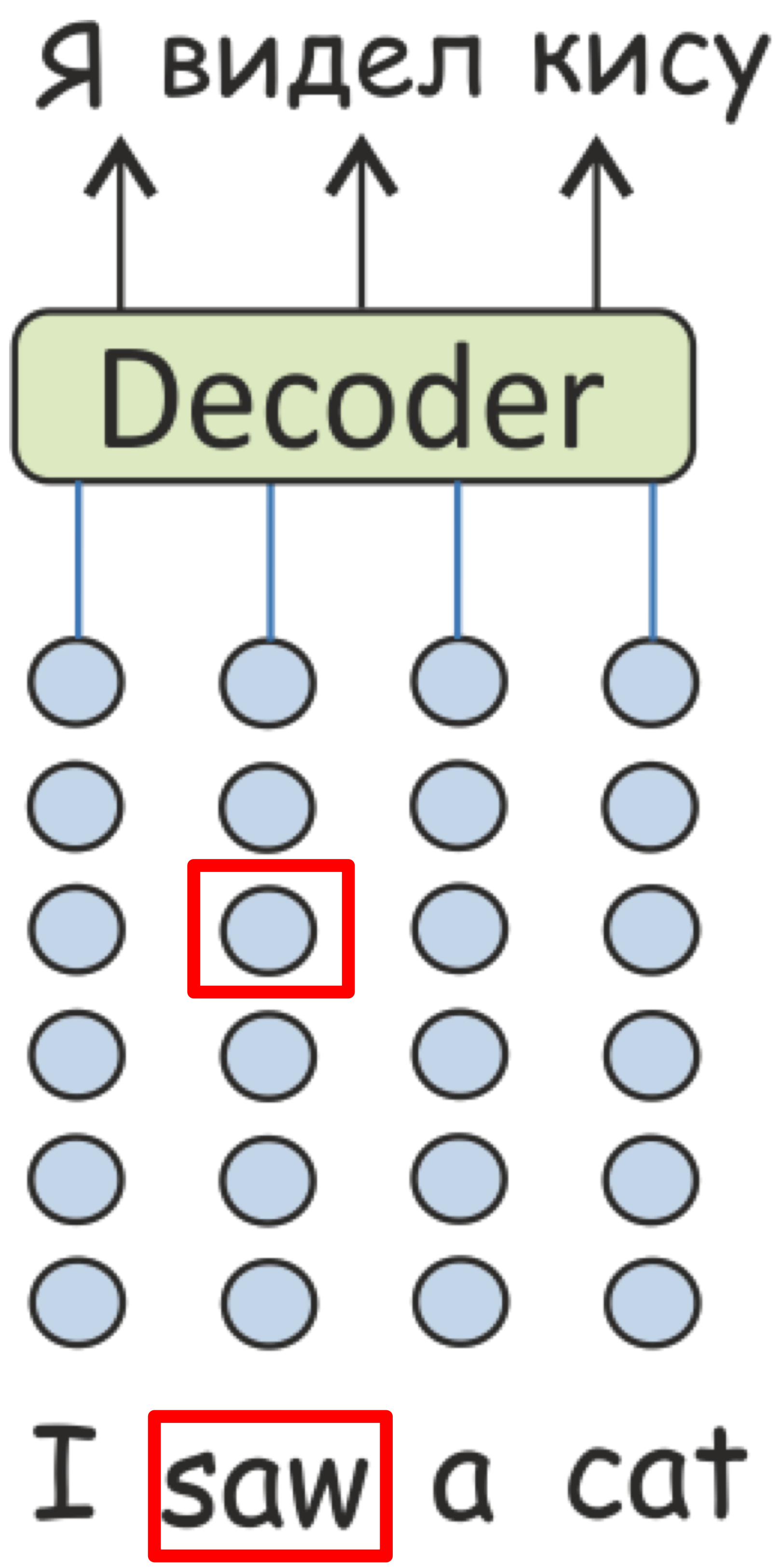
- the nature of changes a token representation undergoes, from layer to layer
- the process of interactions and relationships between tokens
- the type of information which gets lost and acquired by a token representation in these interactions

MI between an input token and a representation

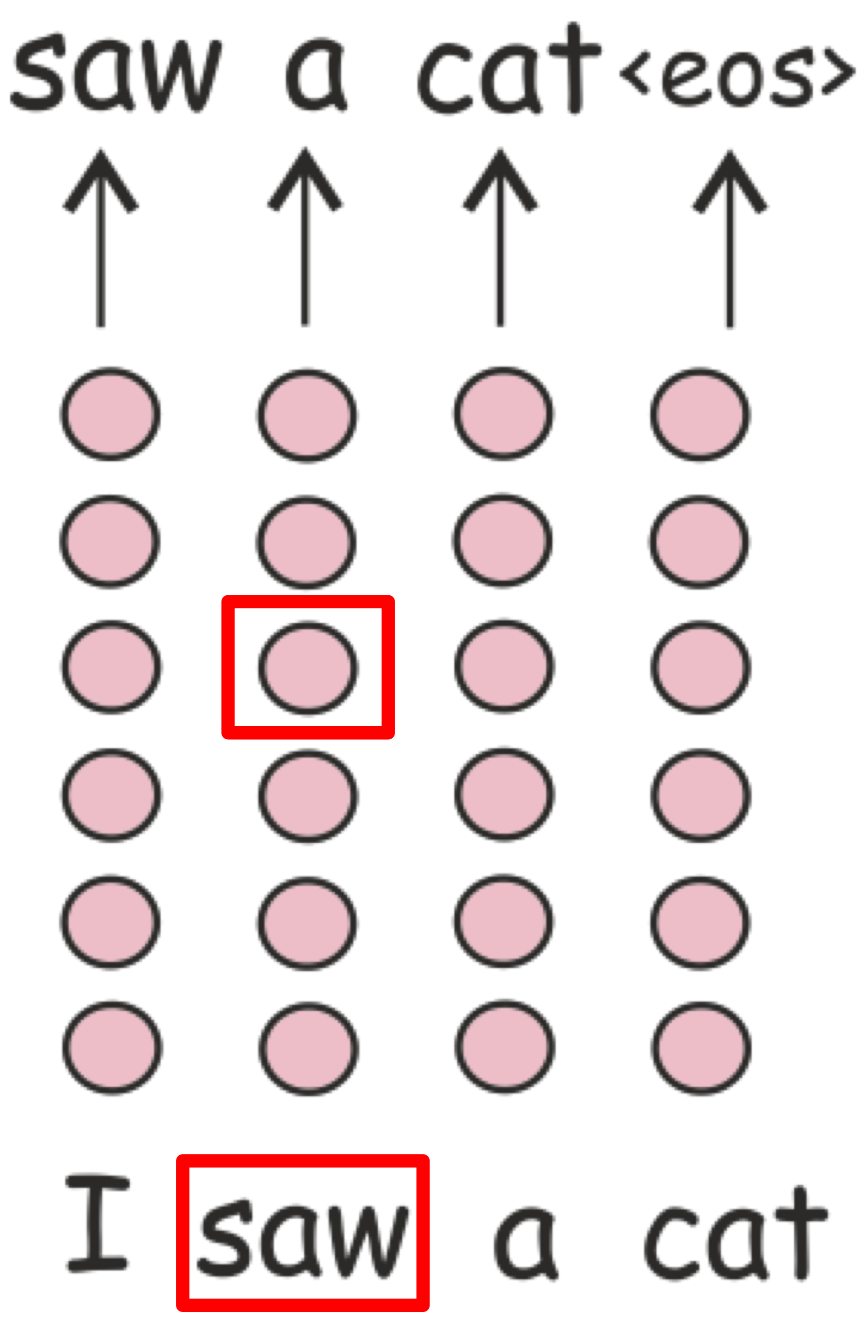


MI between an input token and a representation

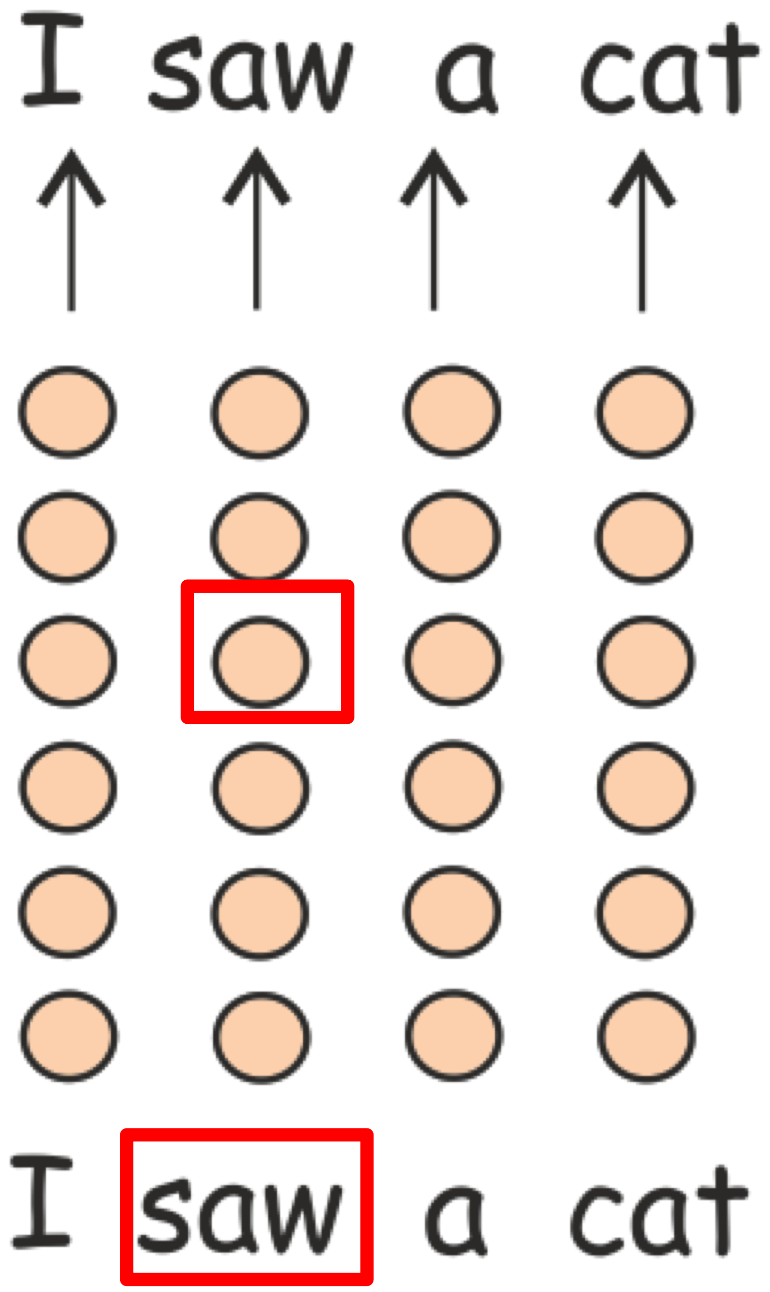
MT



LM

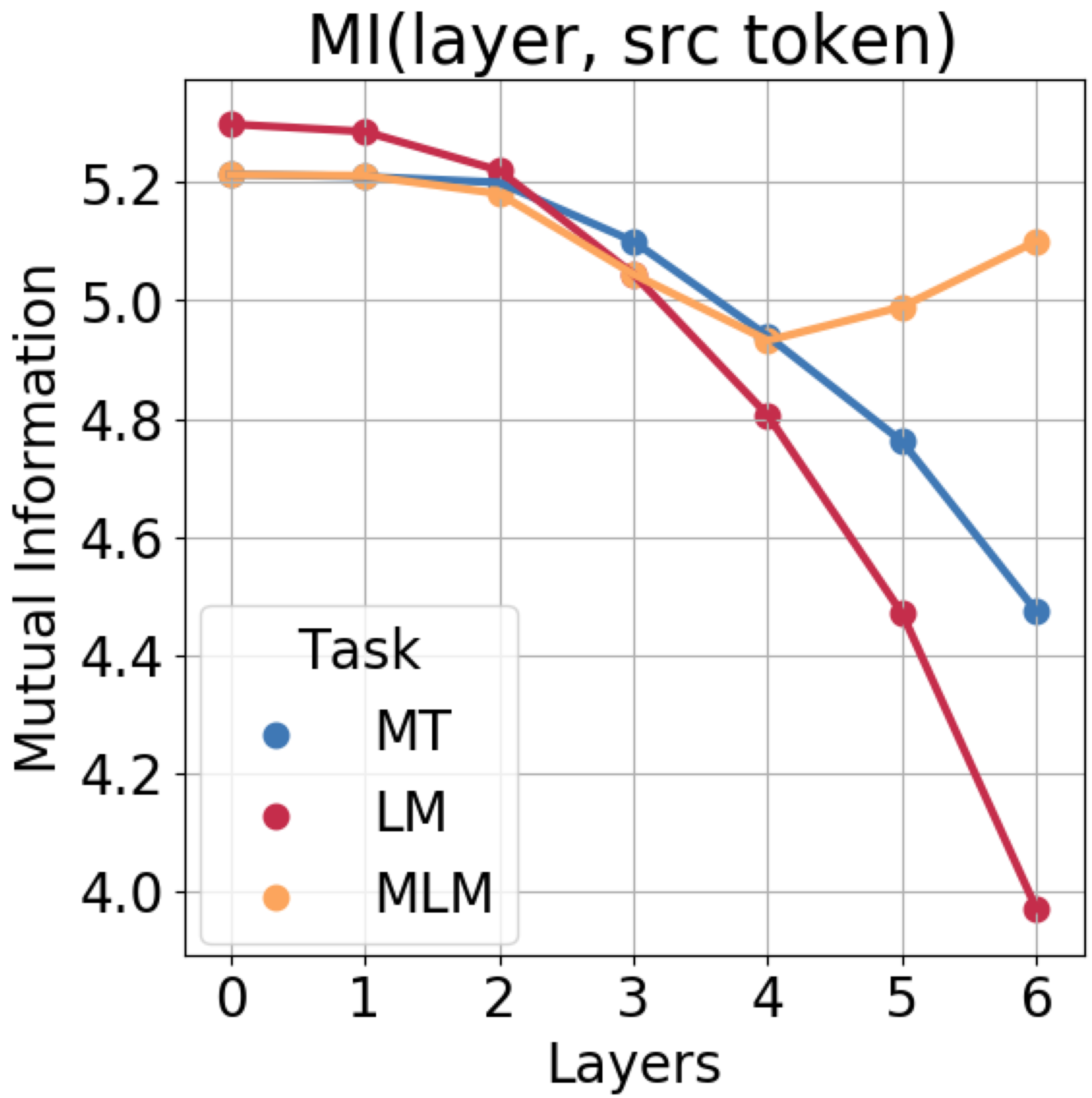


MLM

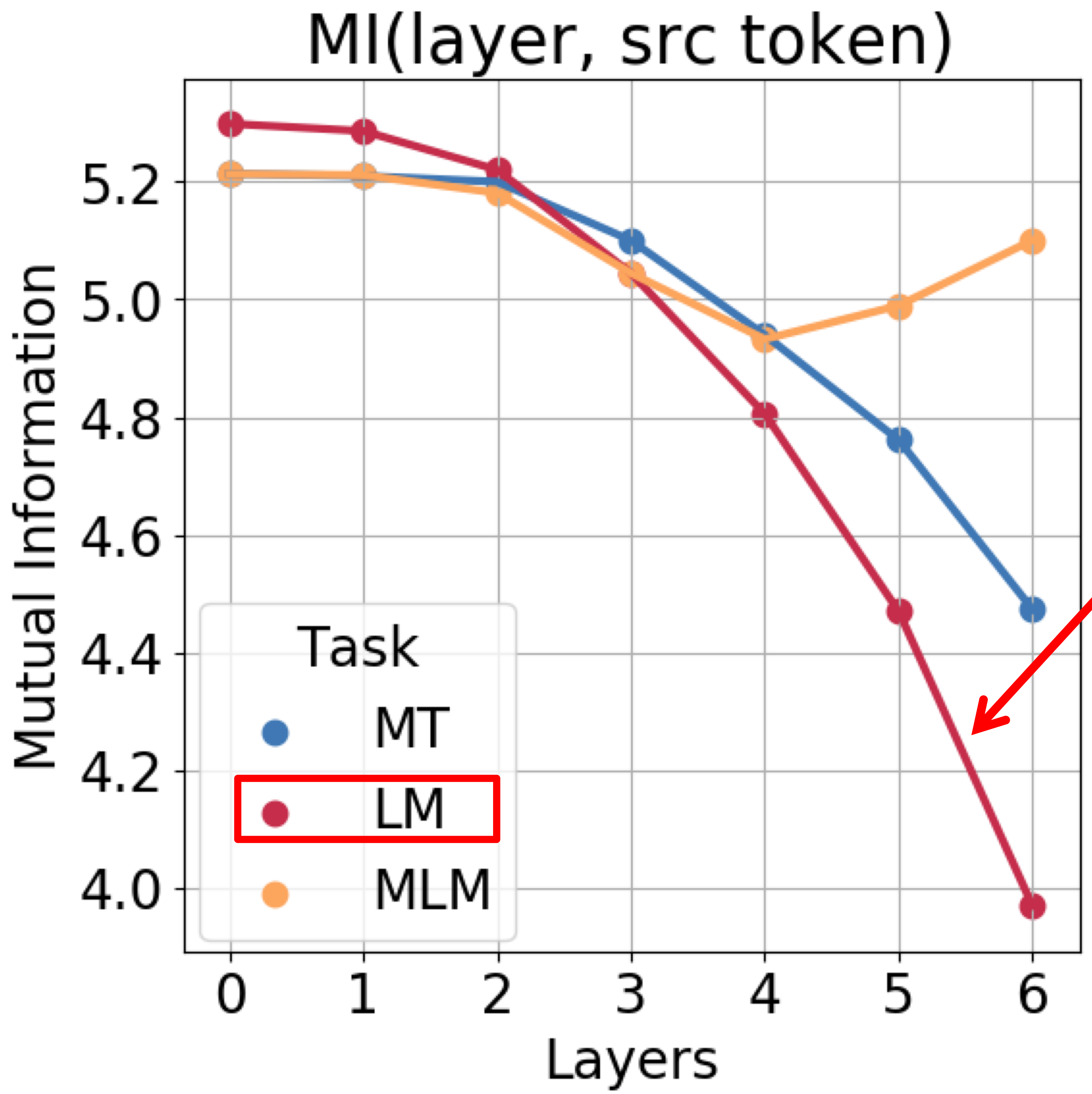


As at test time:
No masking

MI between an input token and a representation

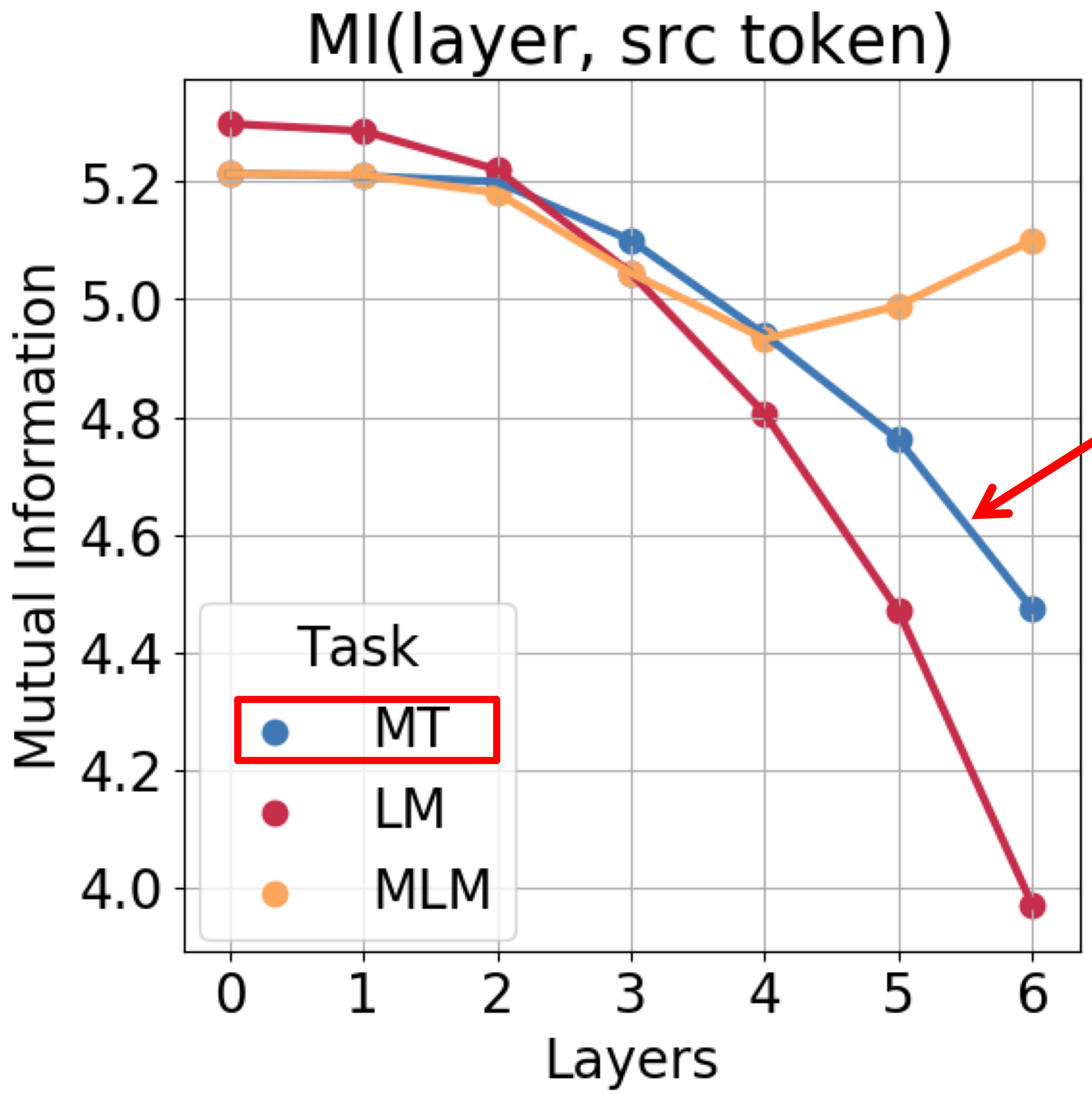


MI between an input token and a representation



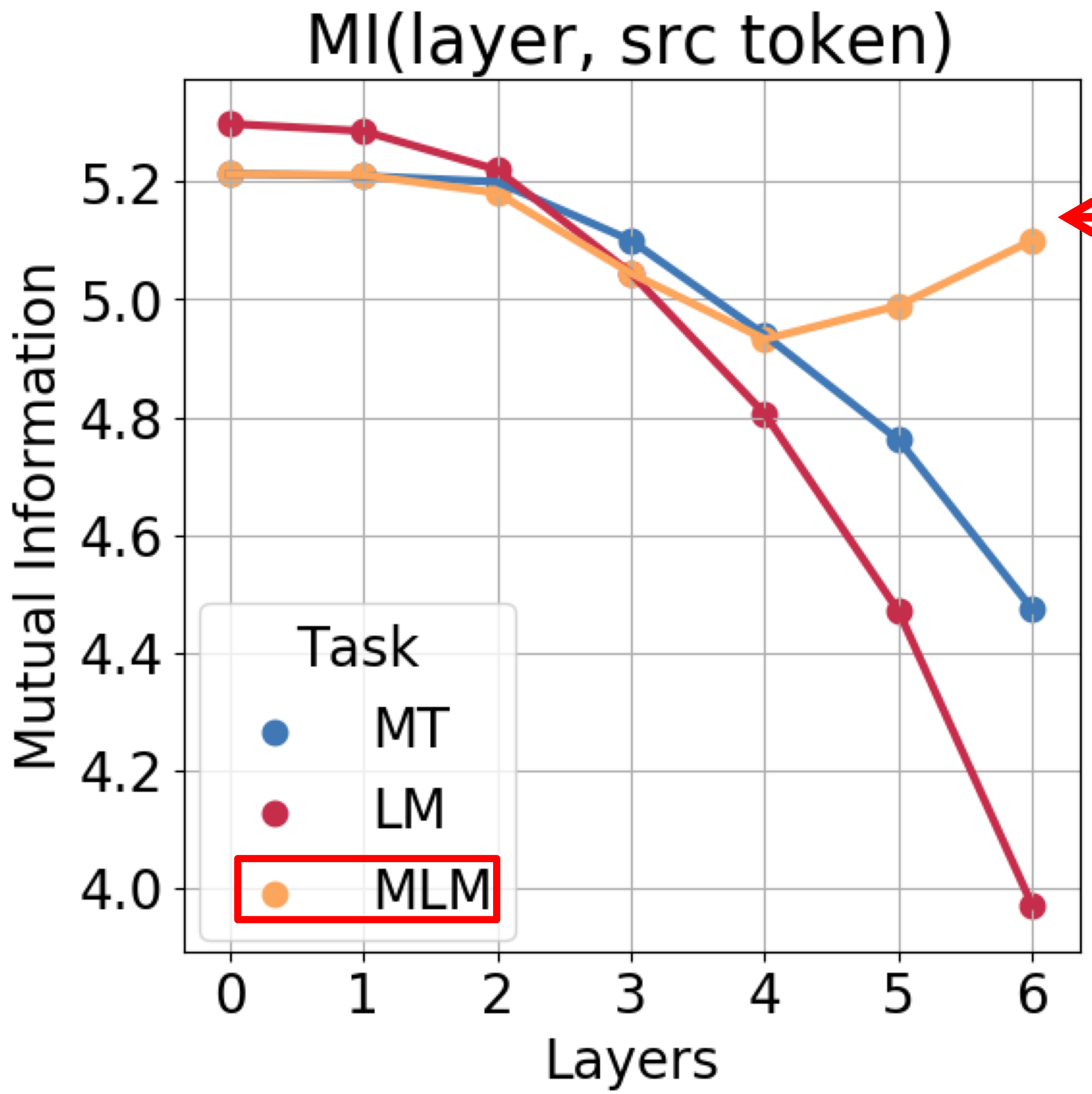
LM loses information about the current input token

MI between an input token and a representation



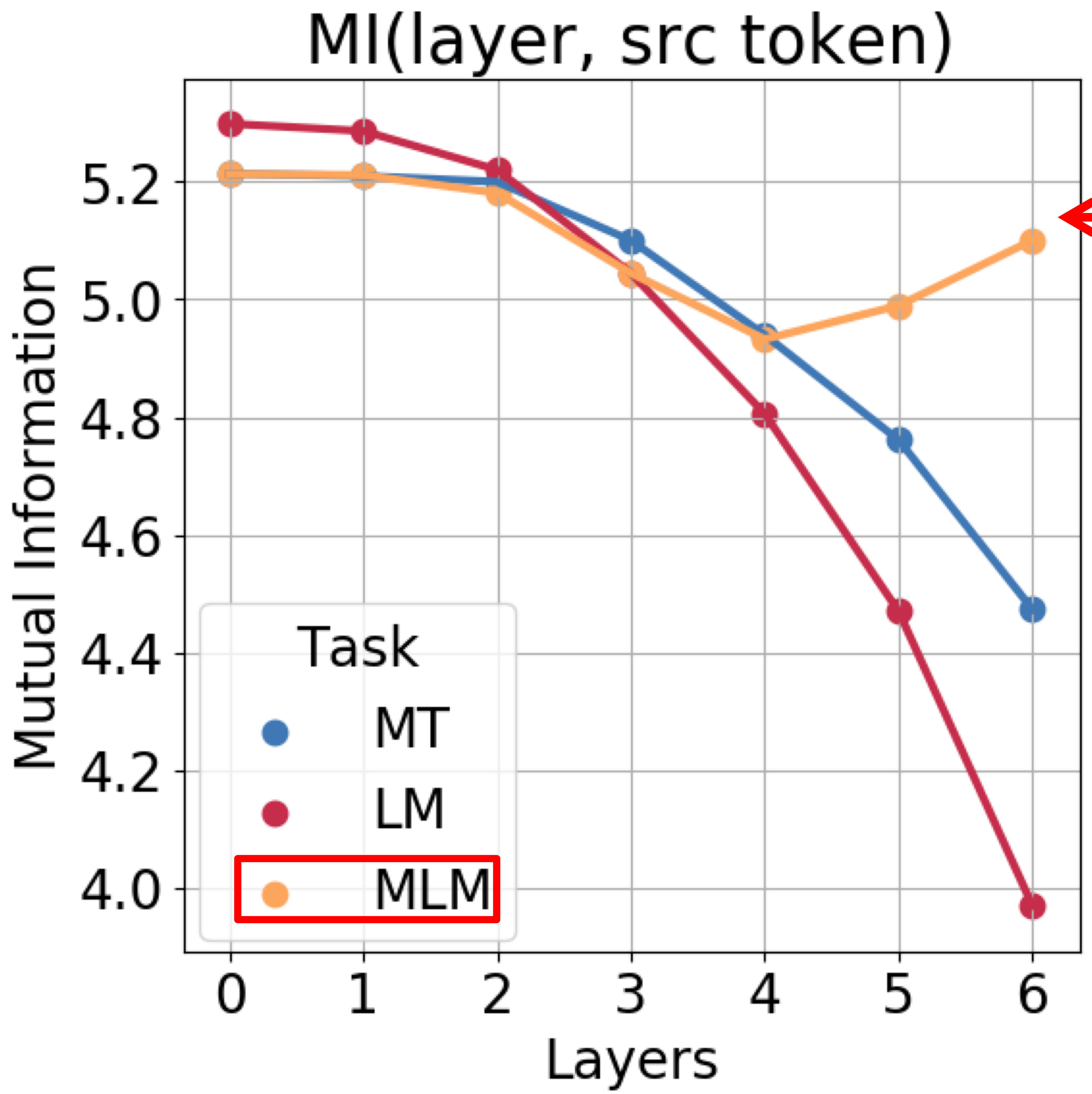
For MT, the behavior is similar, but to lesser extent

MI between an input token and a representation



For MLM, the information about input token gets lost, then recovered

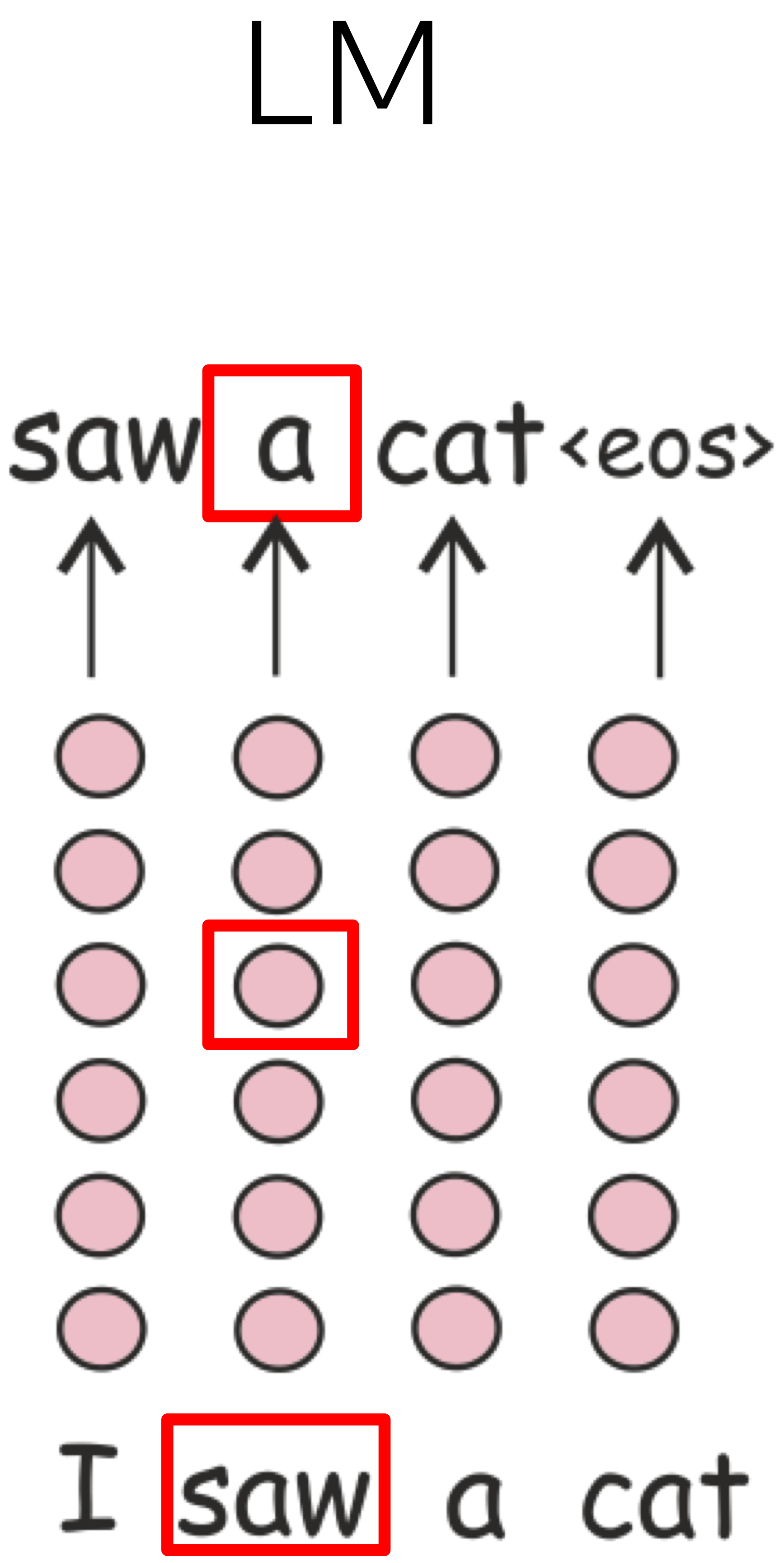
MI between an input token and a representation

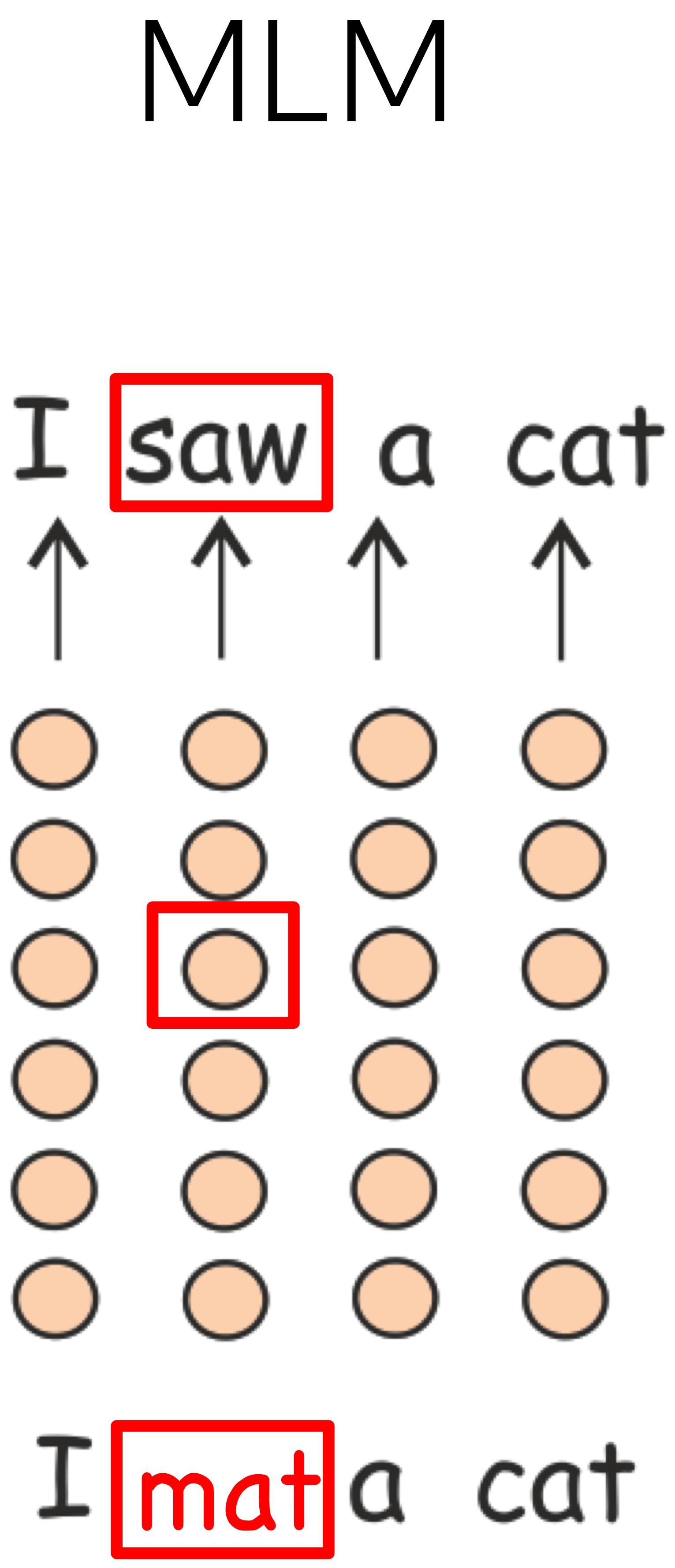


For MLM, the information about input token gets lost, then recovered

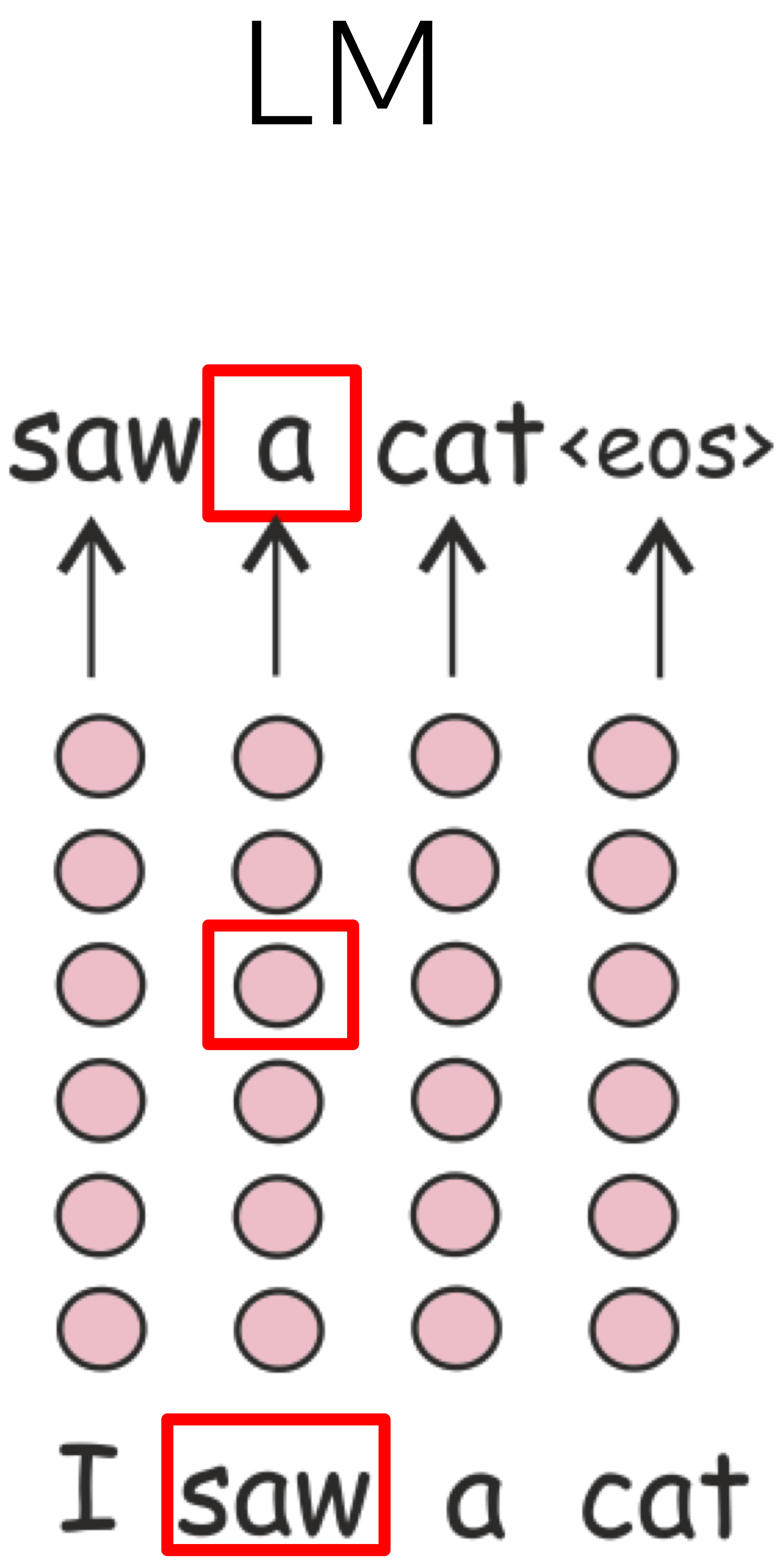
<u>Two stages:</u>
'context encoding' and
'token reconstruction'

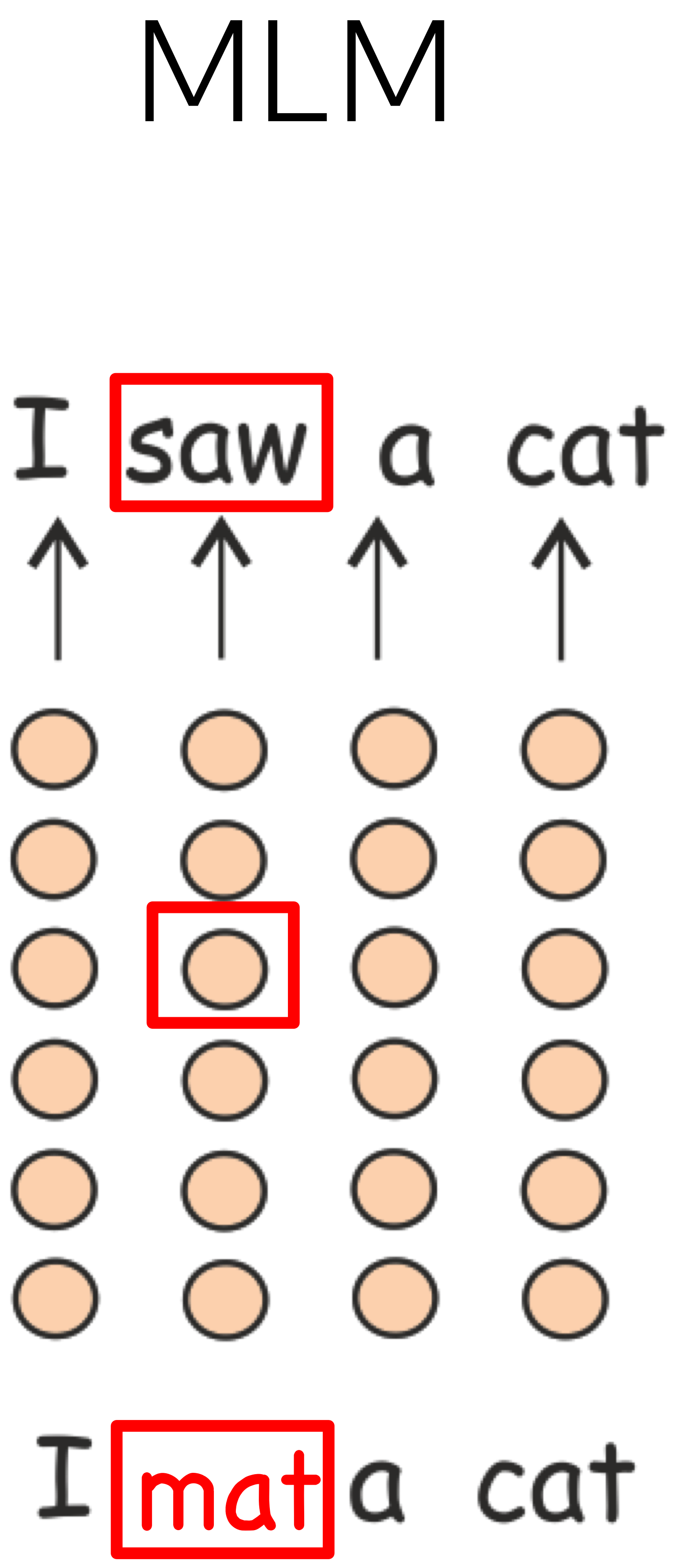
MI between a representation and both input and output





MI between a representation and both input and output

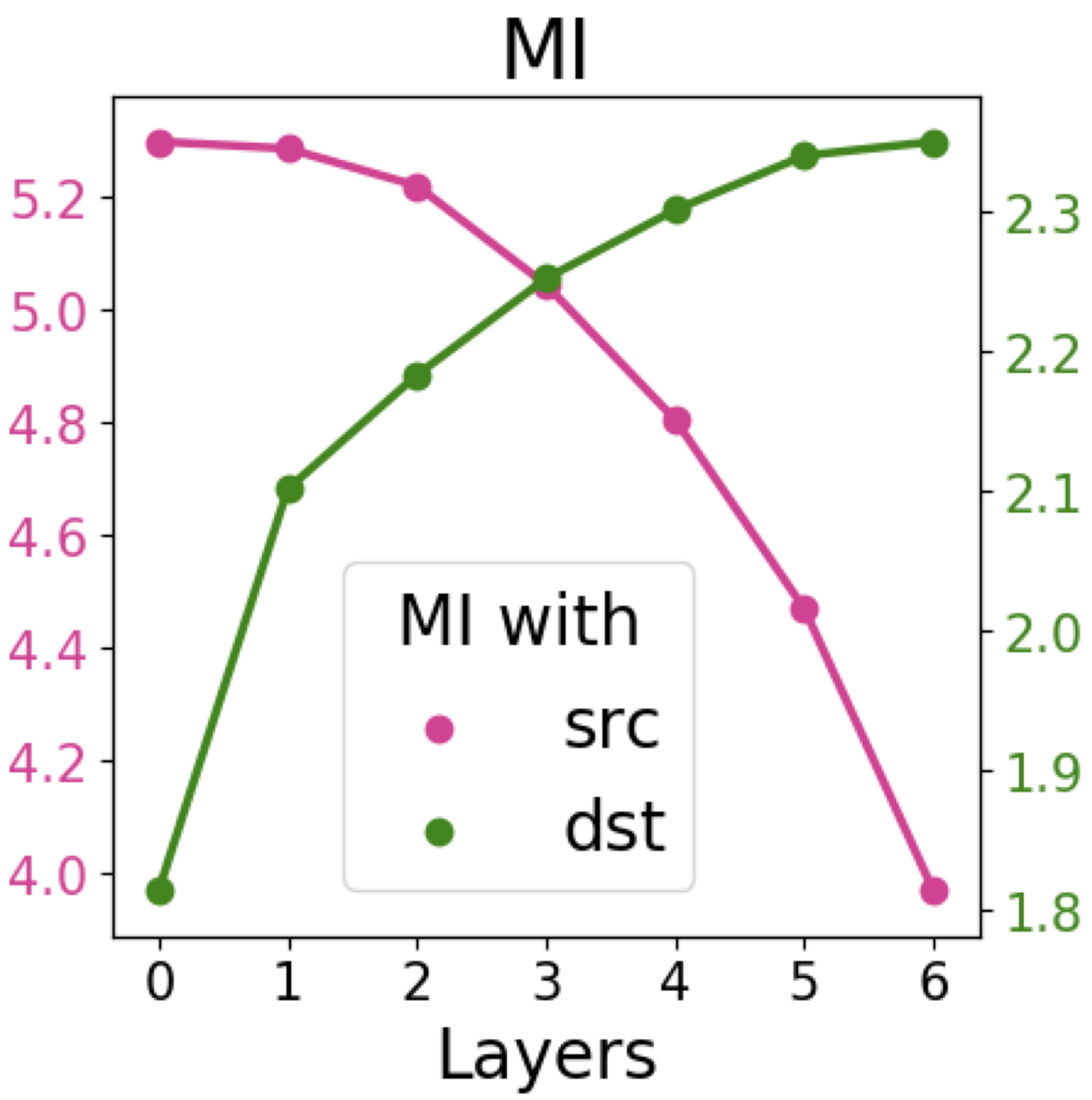




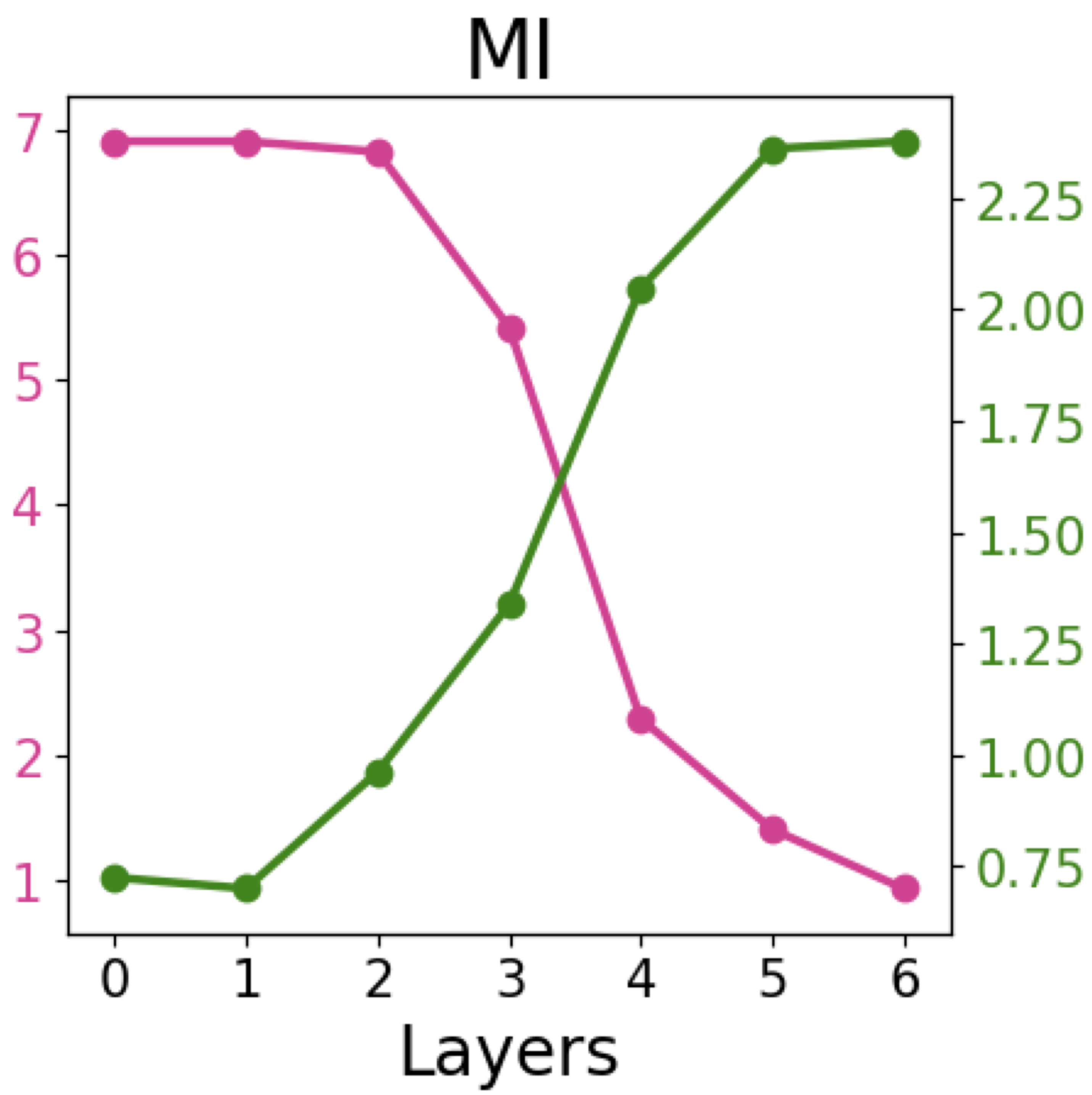
As in training:
With masking and replacing

MI with both input and output tokens

LM



MLM



For MLM:
'context encoding' and 'token prediction' stages

Plan

- Evolution of representations of individual tokens
- Training objectives: LM, MLM, MT
- ”Puzzles” from previous work
- The Information-Bottleneck: our point of view
- Experiments
    - Information Bottleneck for token representations
    - ...

Plan

- Evolution of representations of individual tokens
- Training objectives: LM, MLM, MT
- ”Puzzles” from previous work
- The Information-Bottleneck: our point of view
- Experiments
    - Information Bottleneck for token representations
    - Analyzing changes and influences
    - …

Analyzing Changes and Influences

Analyzing Changes and Influences

- how much change is happening in a given layer
- which tokens gain more information from other tokens
- which tokens influence other tokens most

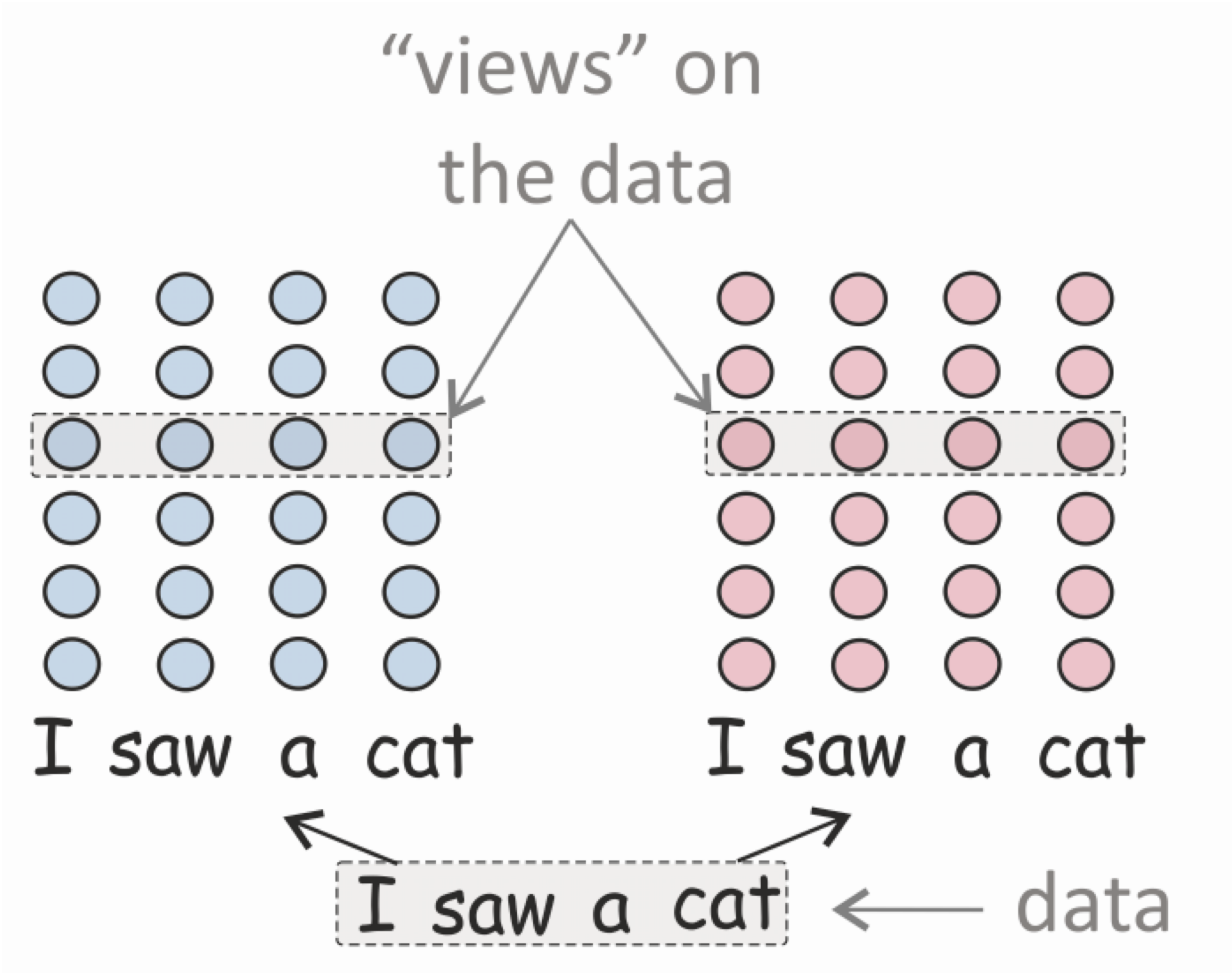
Analyzing Changes and Influences

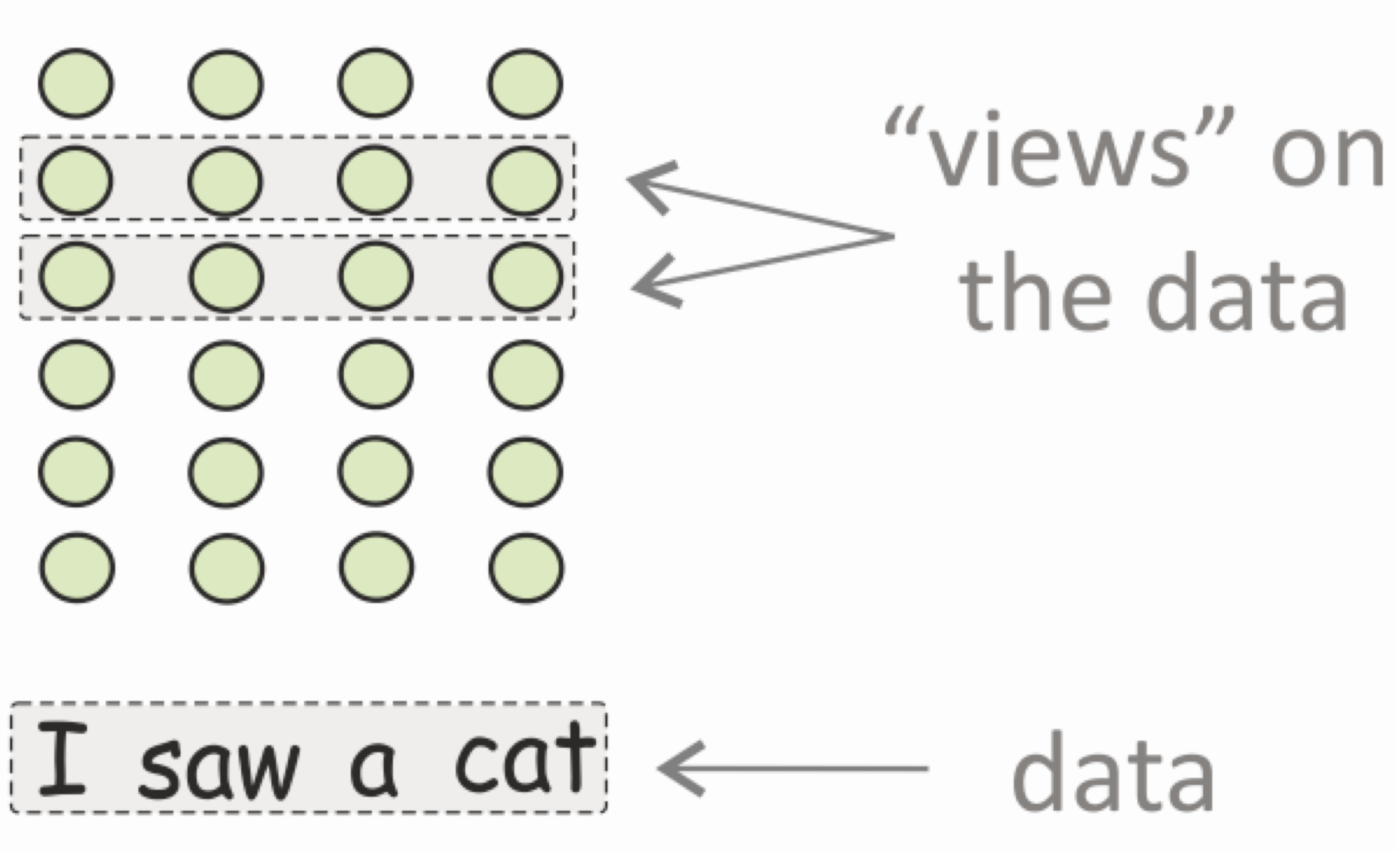
- how much change is happening in a given layer
- which tokens gain more information from other tokens
- which tokens influence other tokens most

→ Comparison between network representations

Views on the data

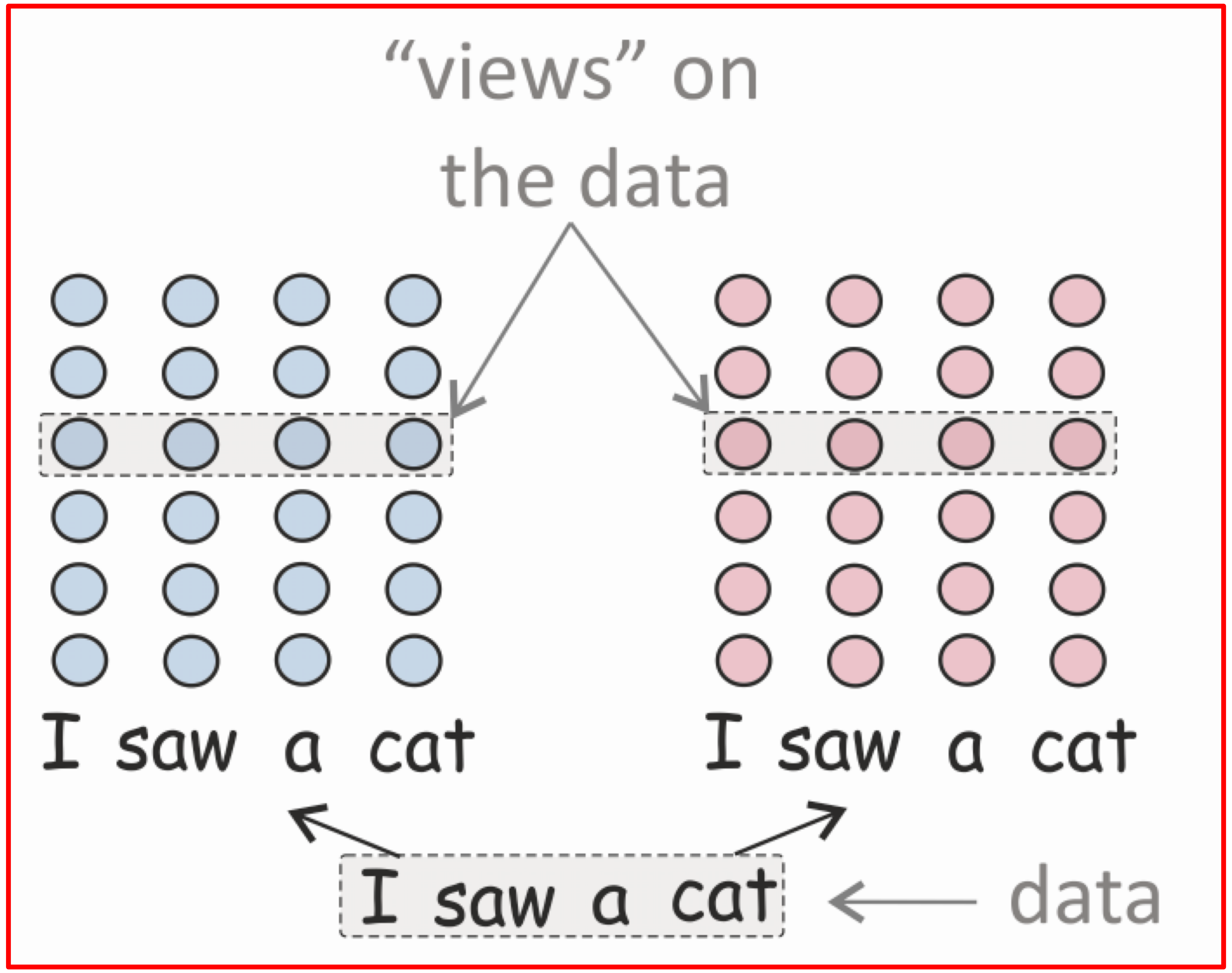
- use PWCCA – a version of canonical correlation analysis (CCA)
- PWCCA measures similarity between pairs of ‘views’ on the data

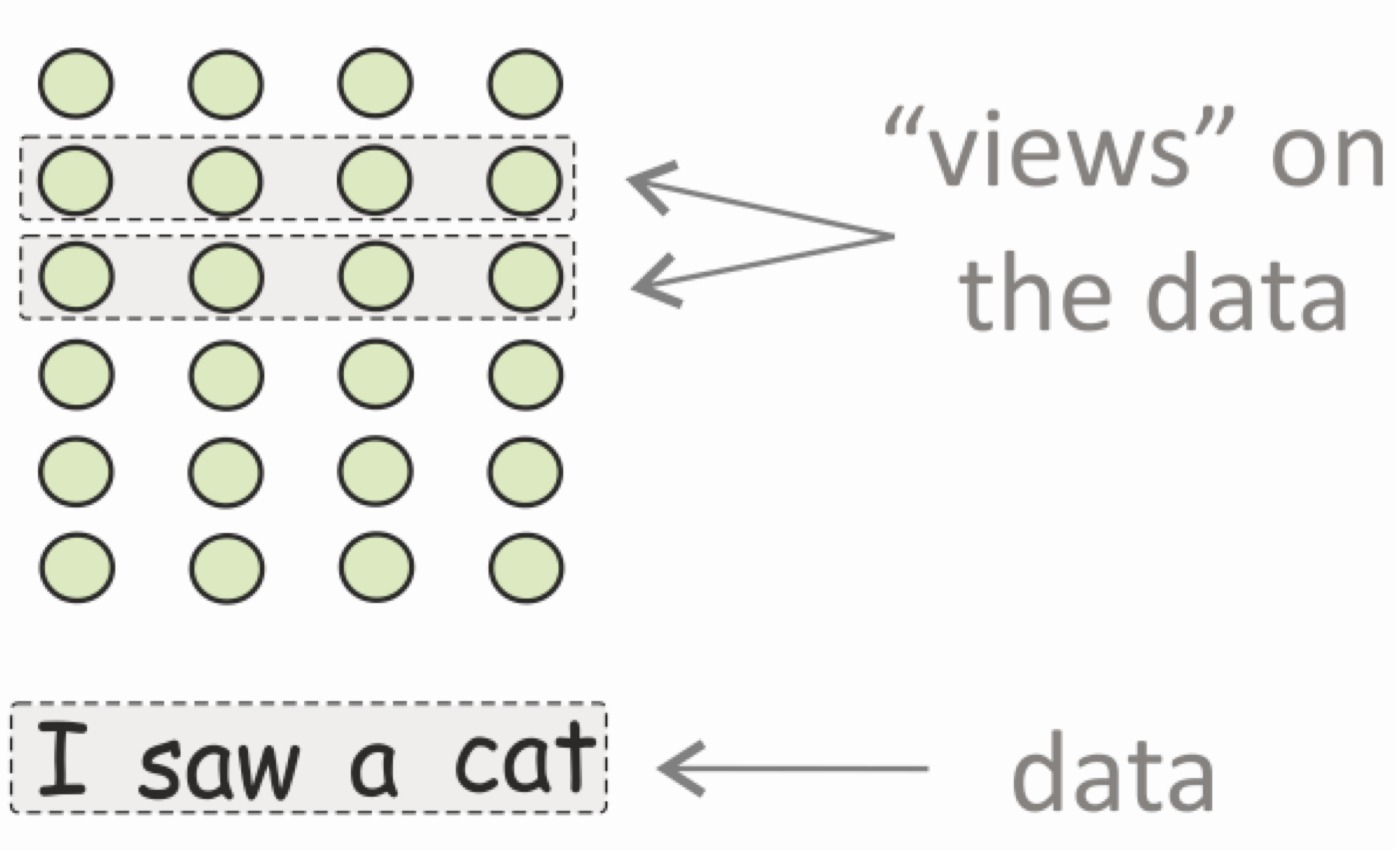




Views on the data

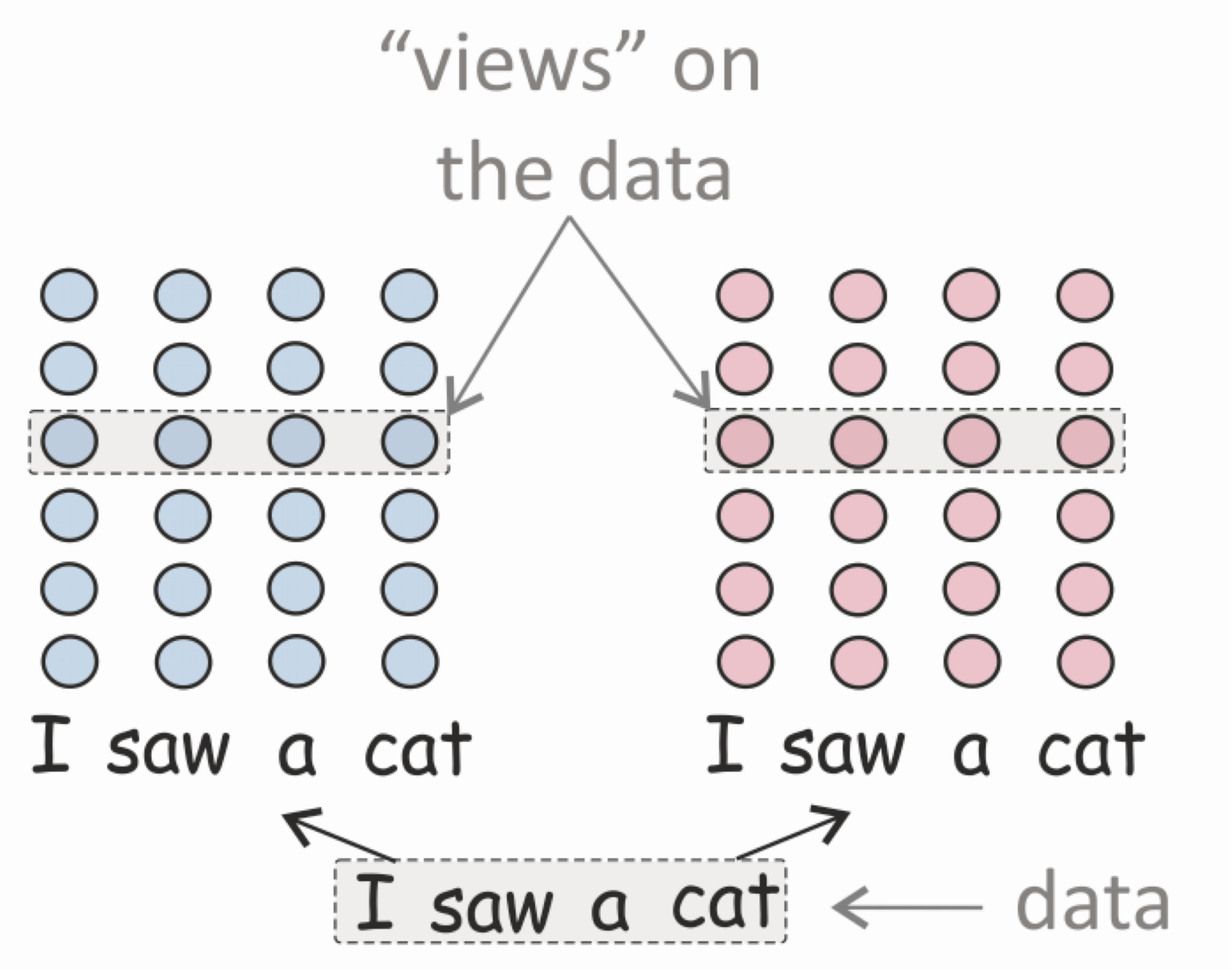
- use PWCCA – a version of canonical correlation analysis (CCA)
- PWCCA measures similarity between pairs of ‘views’ on the data

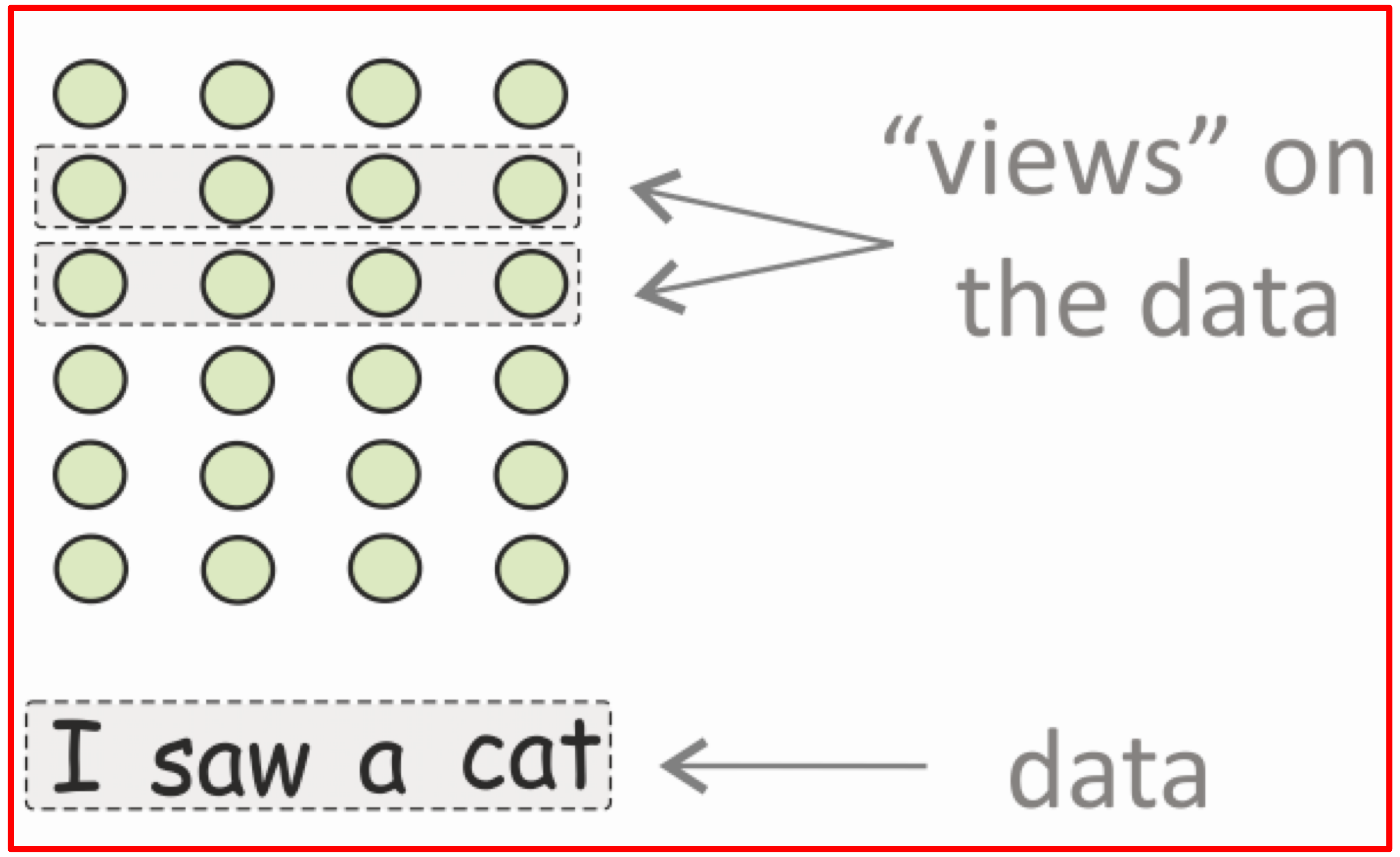




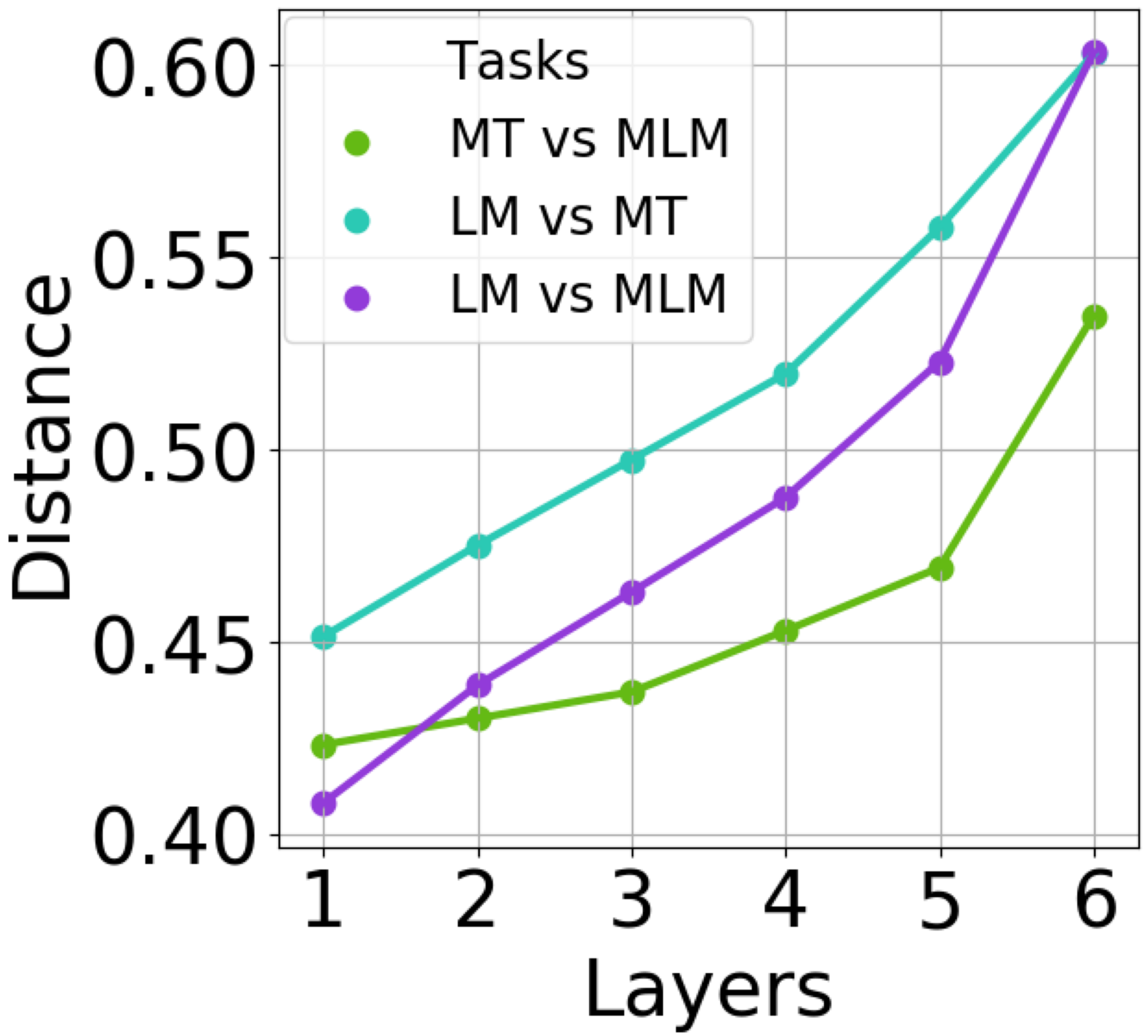
Views on the data

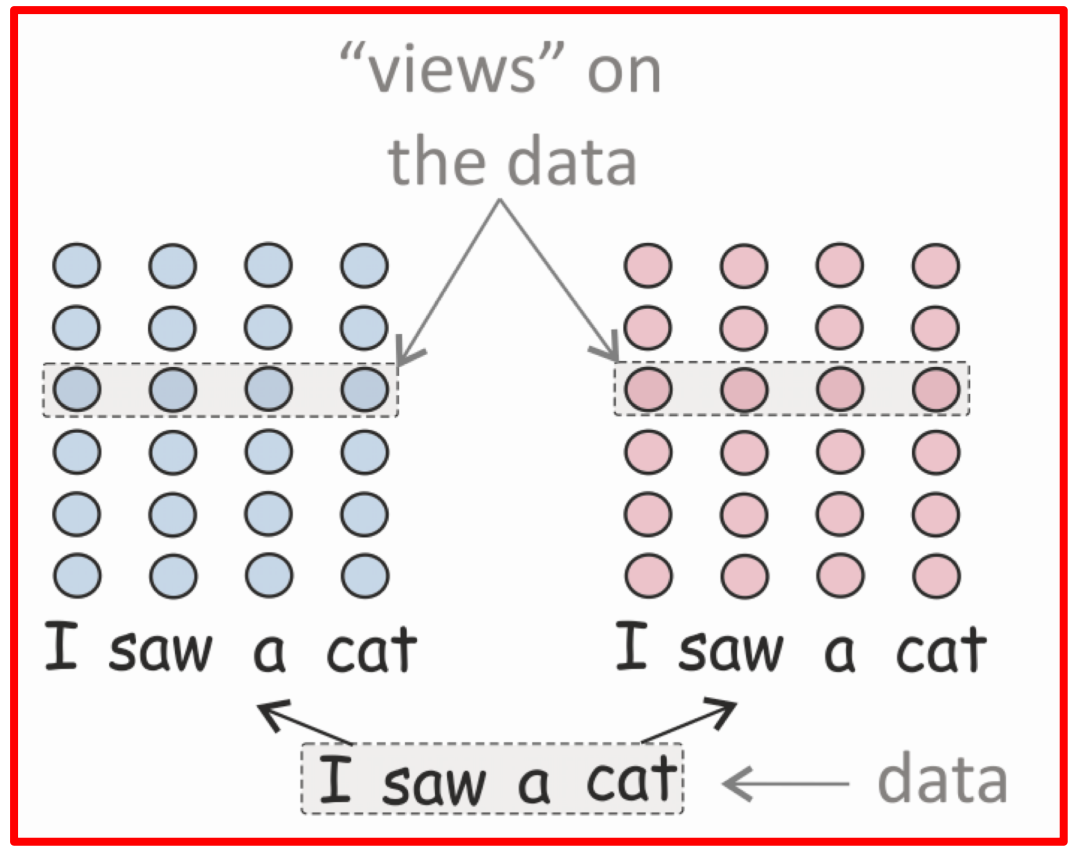
- use PWCCA – a version of canonical correlation analysis (CCA)
- PWCCA measures similarity between pairs of 'views' on the data



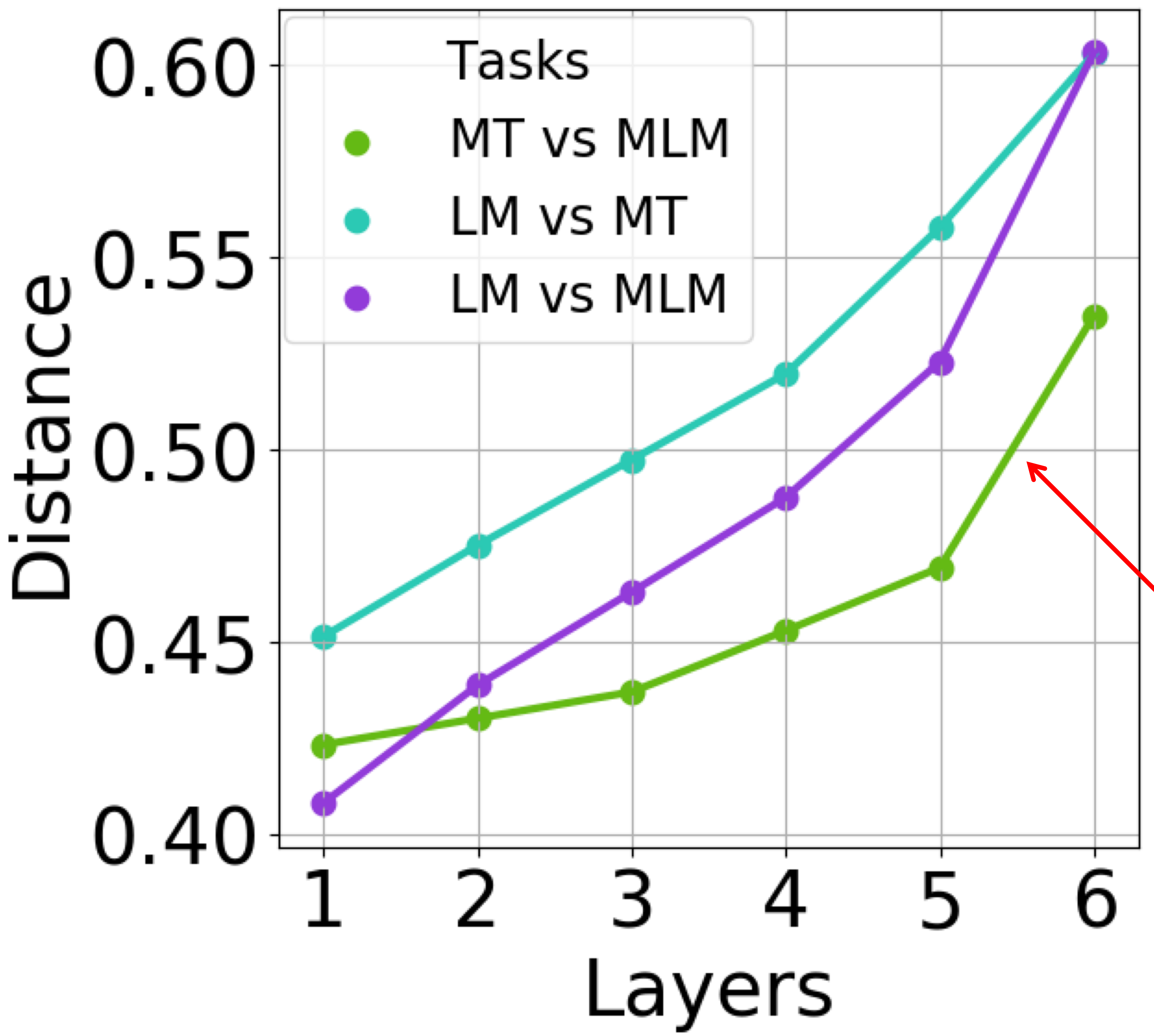


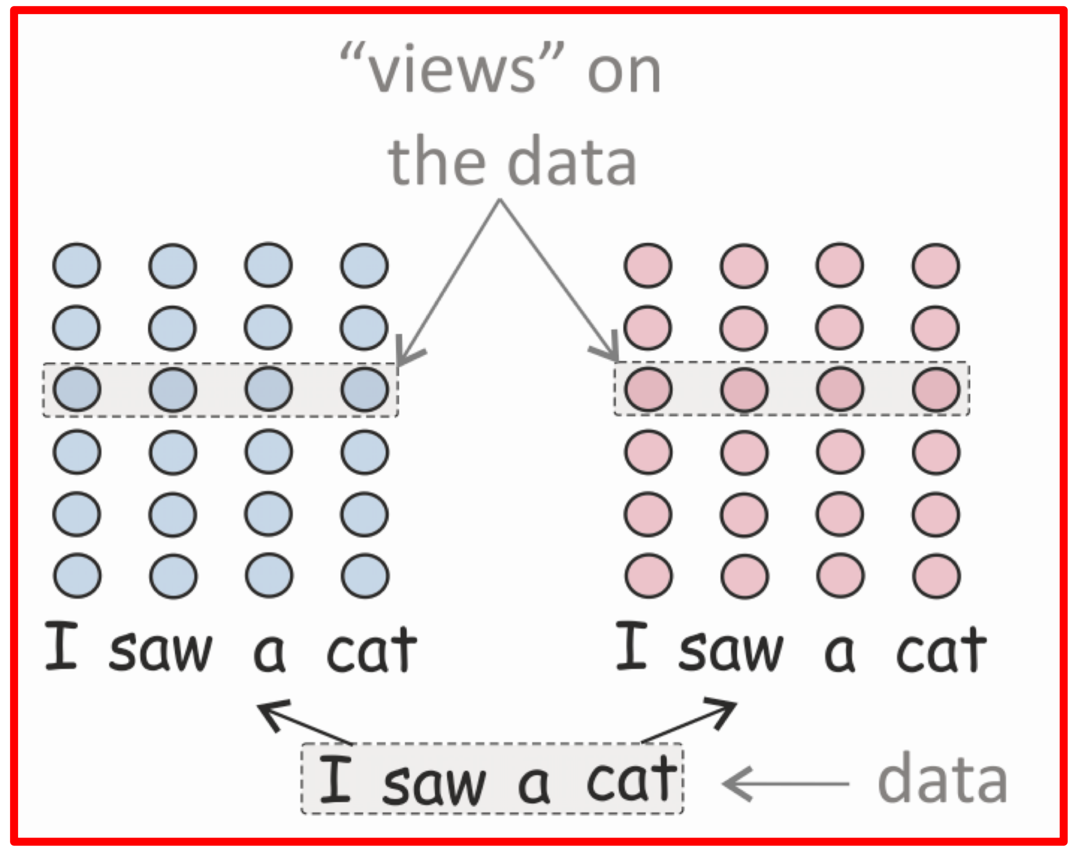
A coarse-grained view: Distance between tasks





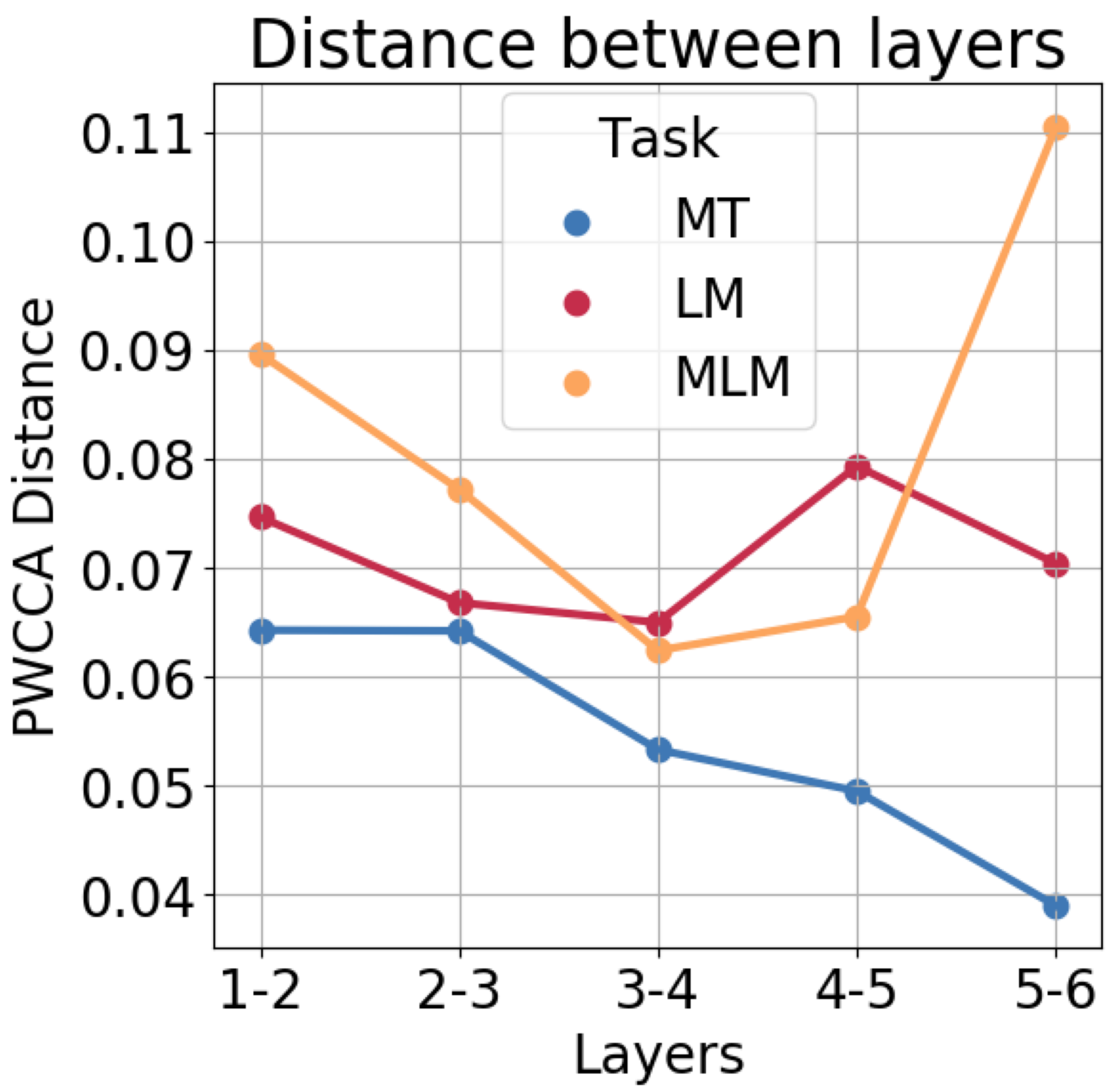
A coarse-grained view: Distance between tasks

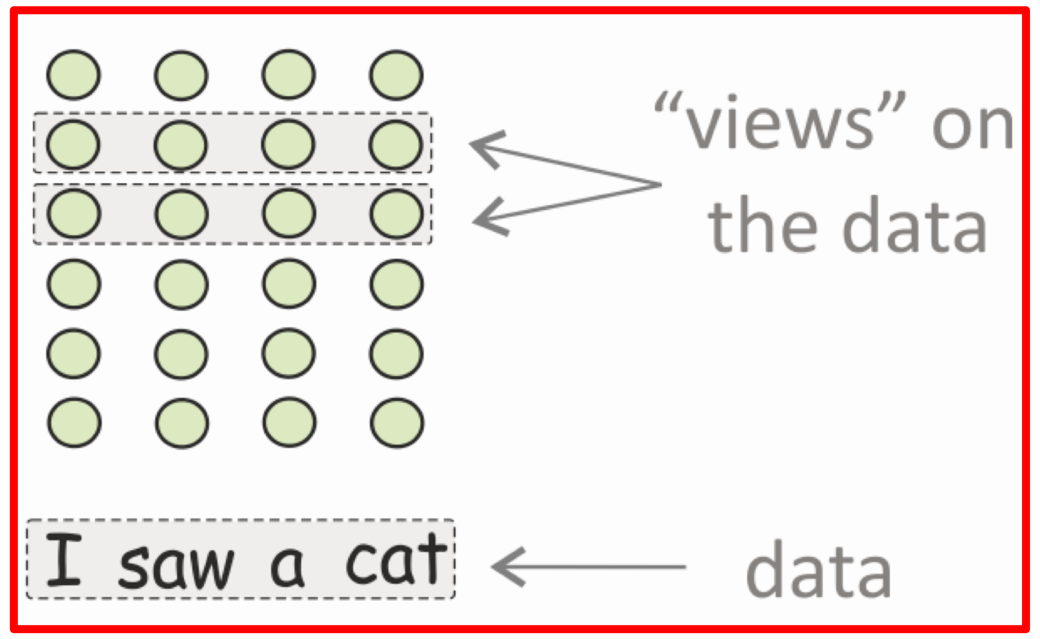




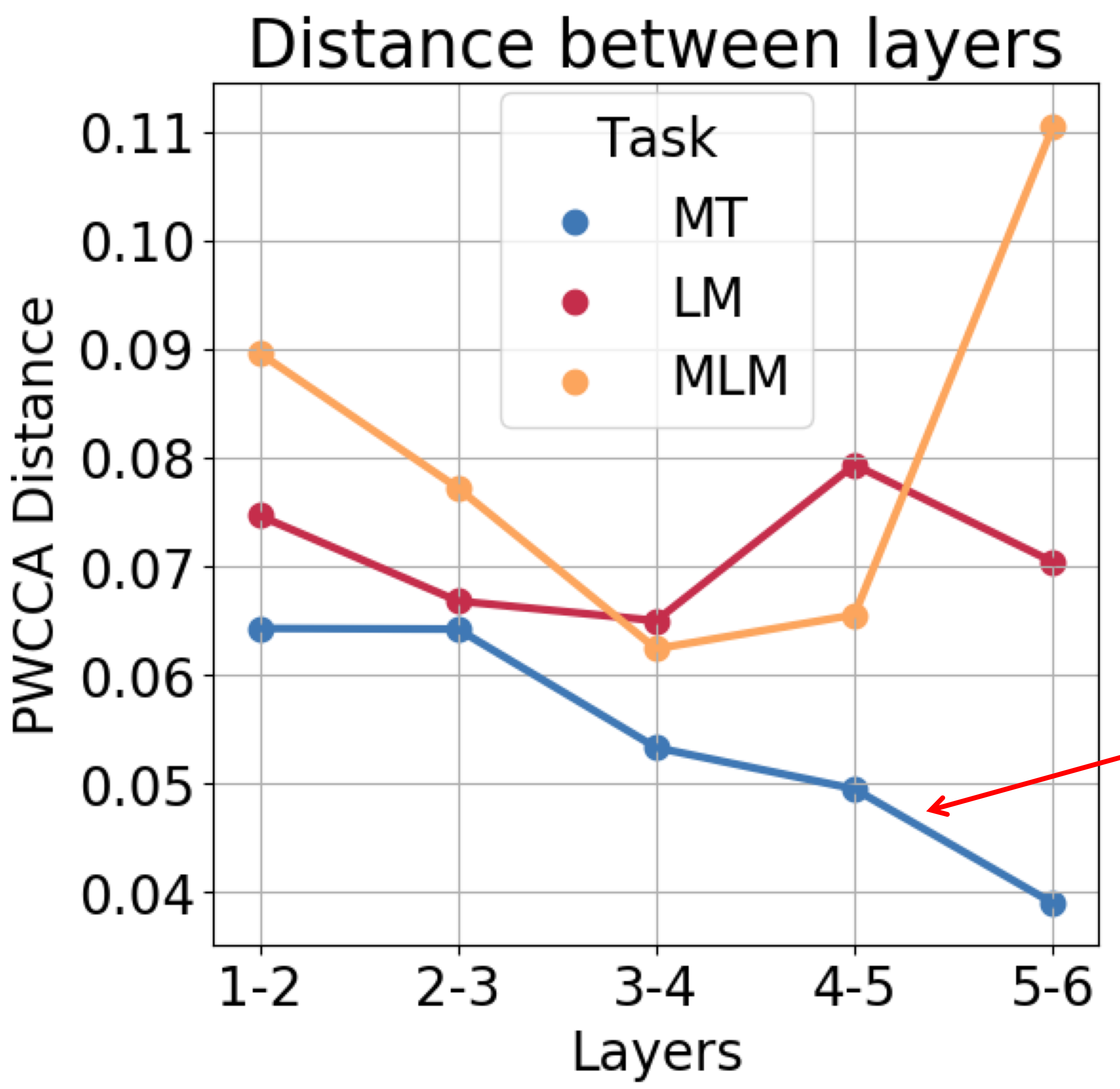
MT and MLM are closer to each other, than they are to LM

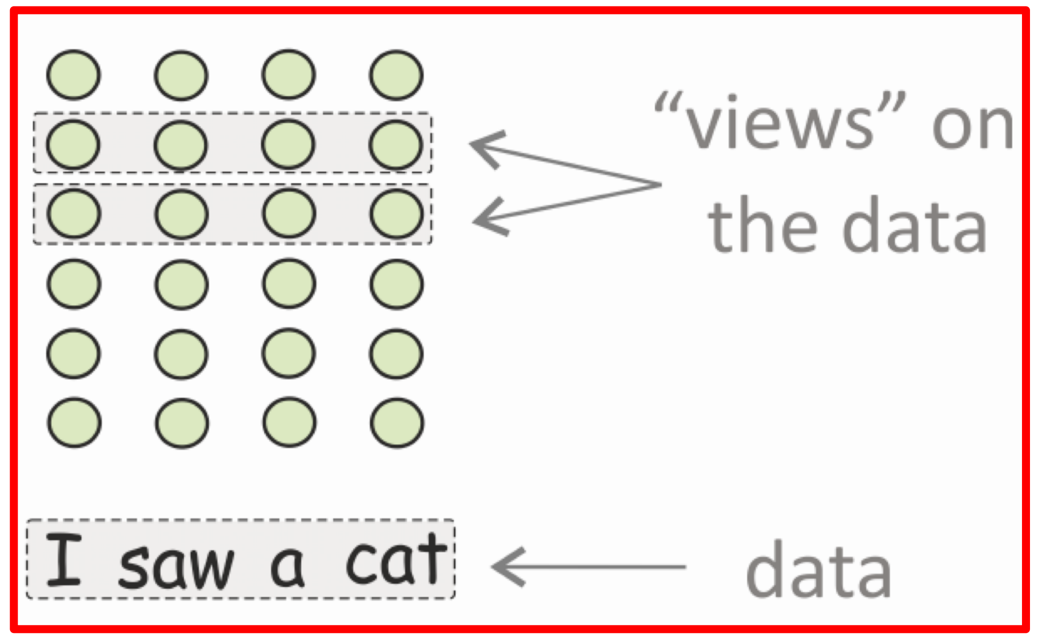
A coarse-grained view: Changes between layers





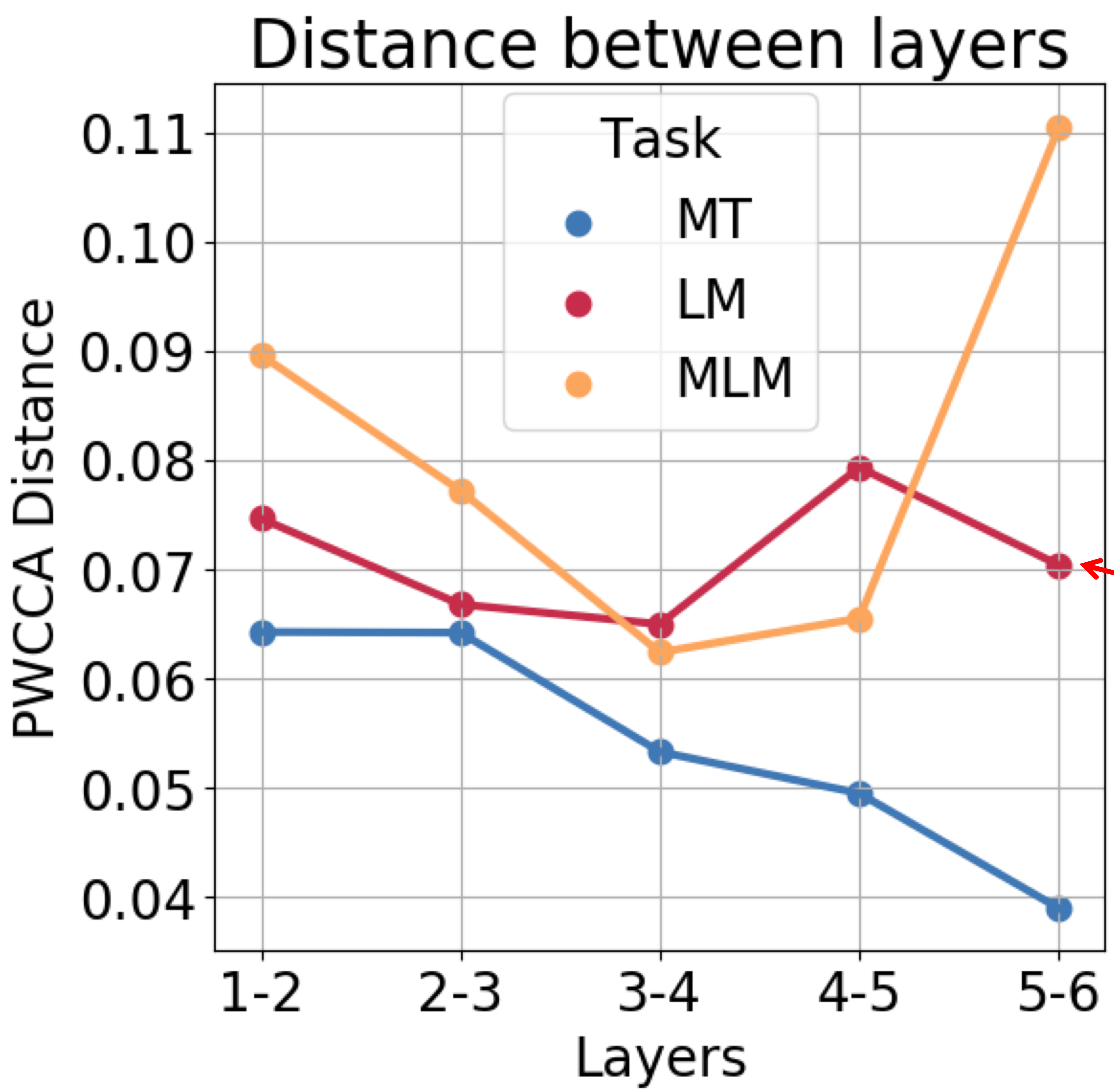
A coarse-grained view: Changes between layers

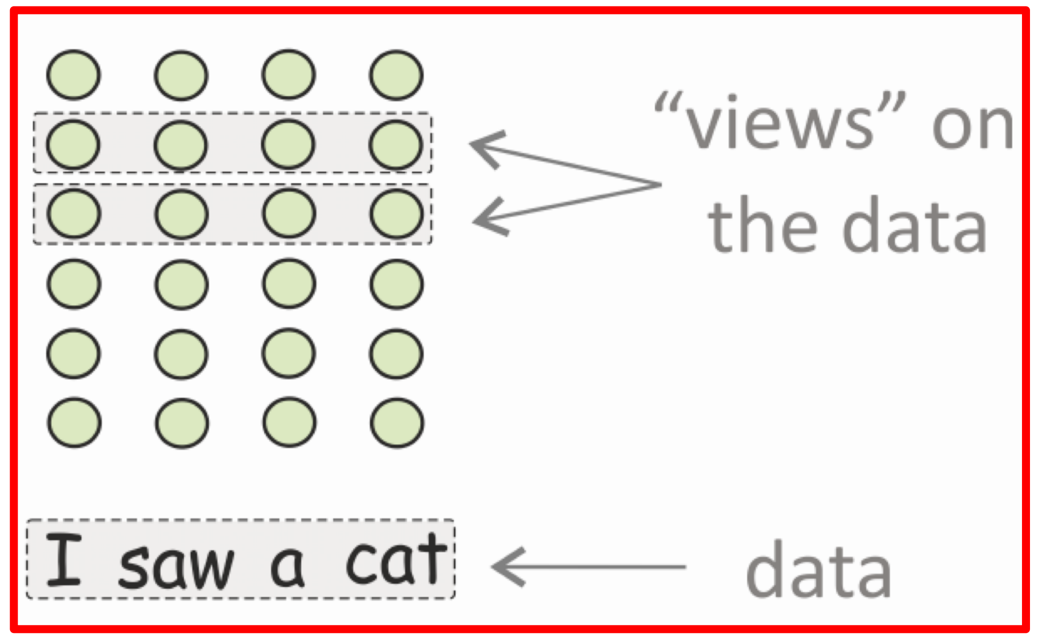




decreasing change for MT

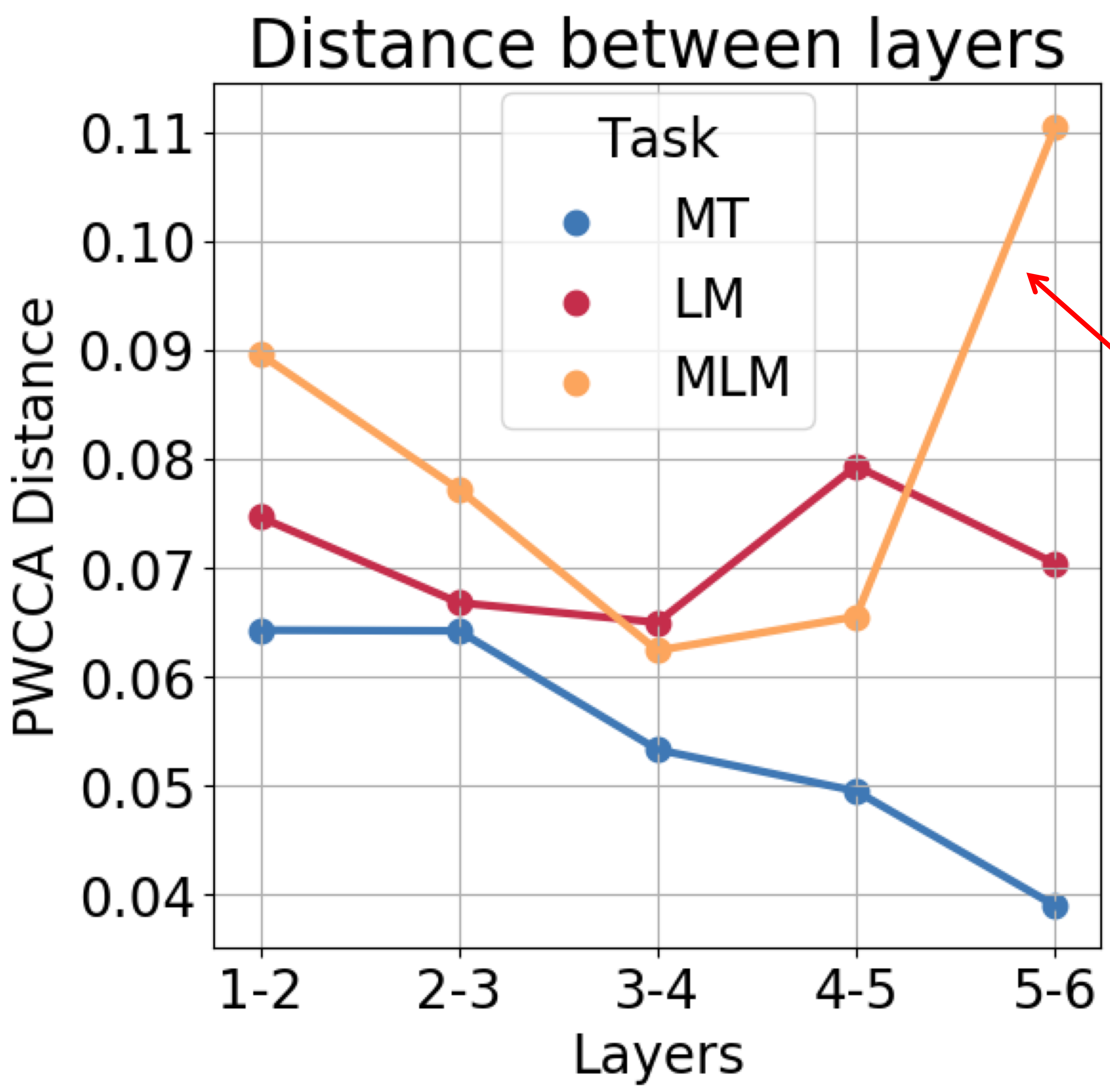
A coarse-grained view: Changes between layers

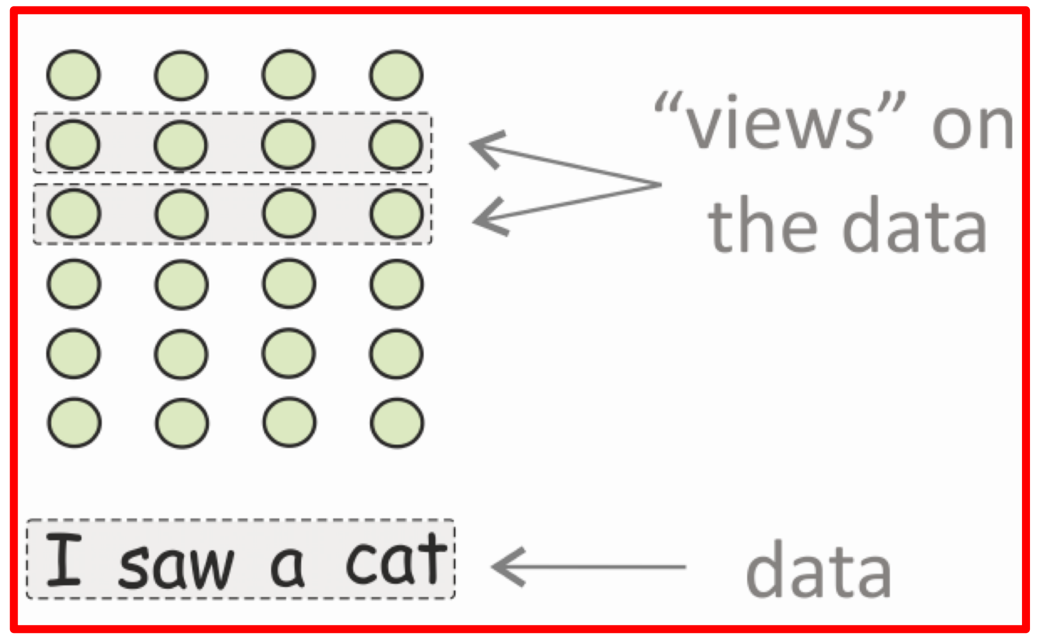




a lot of
change for LM

A coarse-grained view: Changes between layers

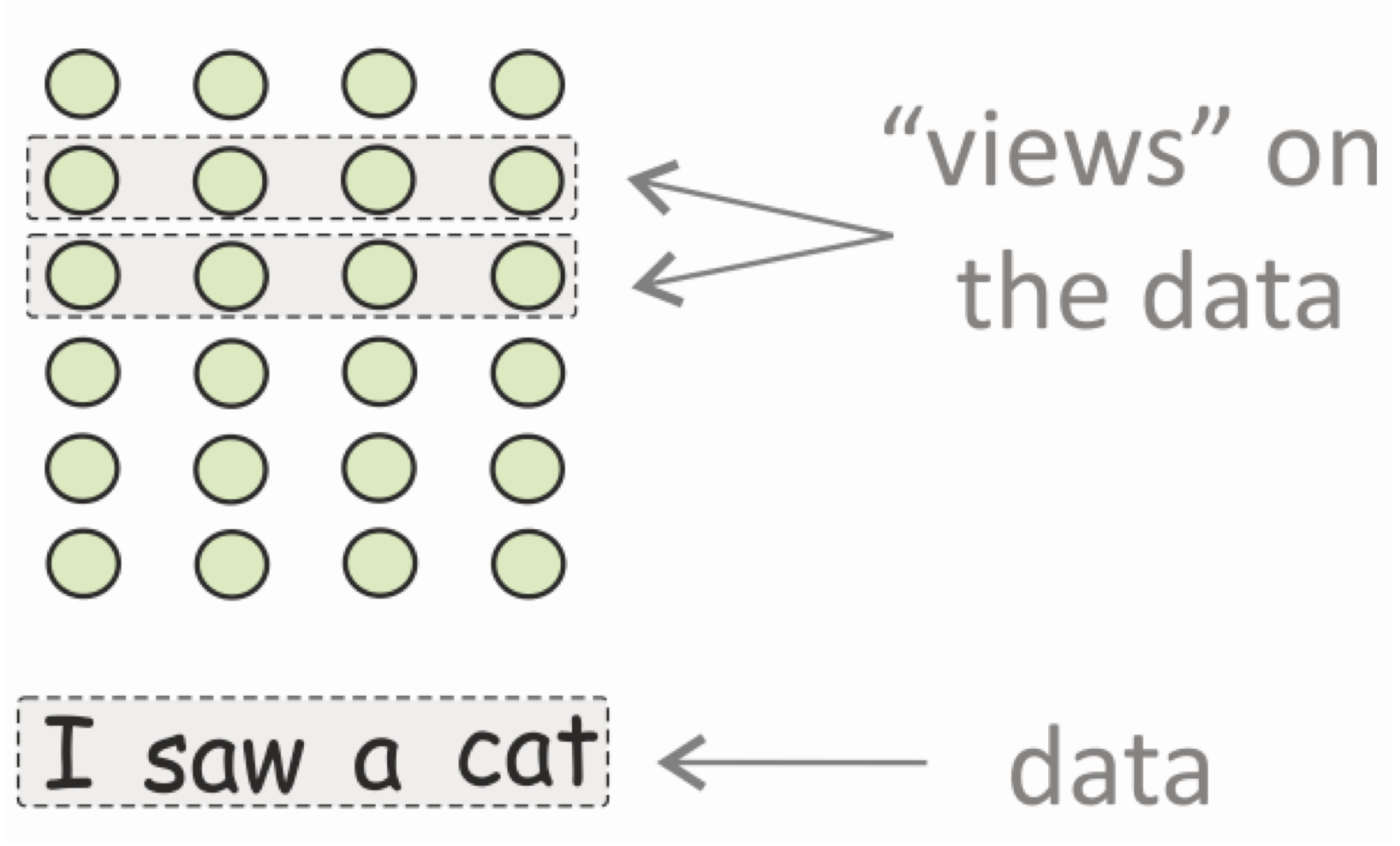


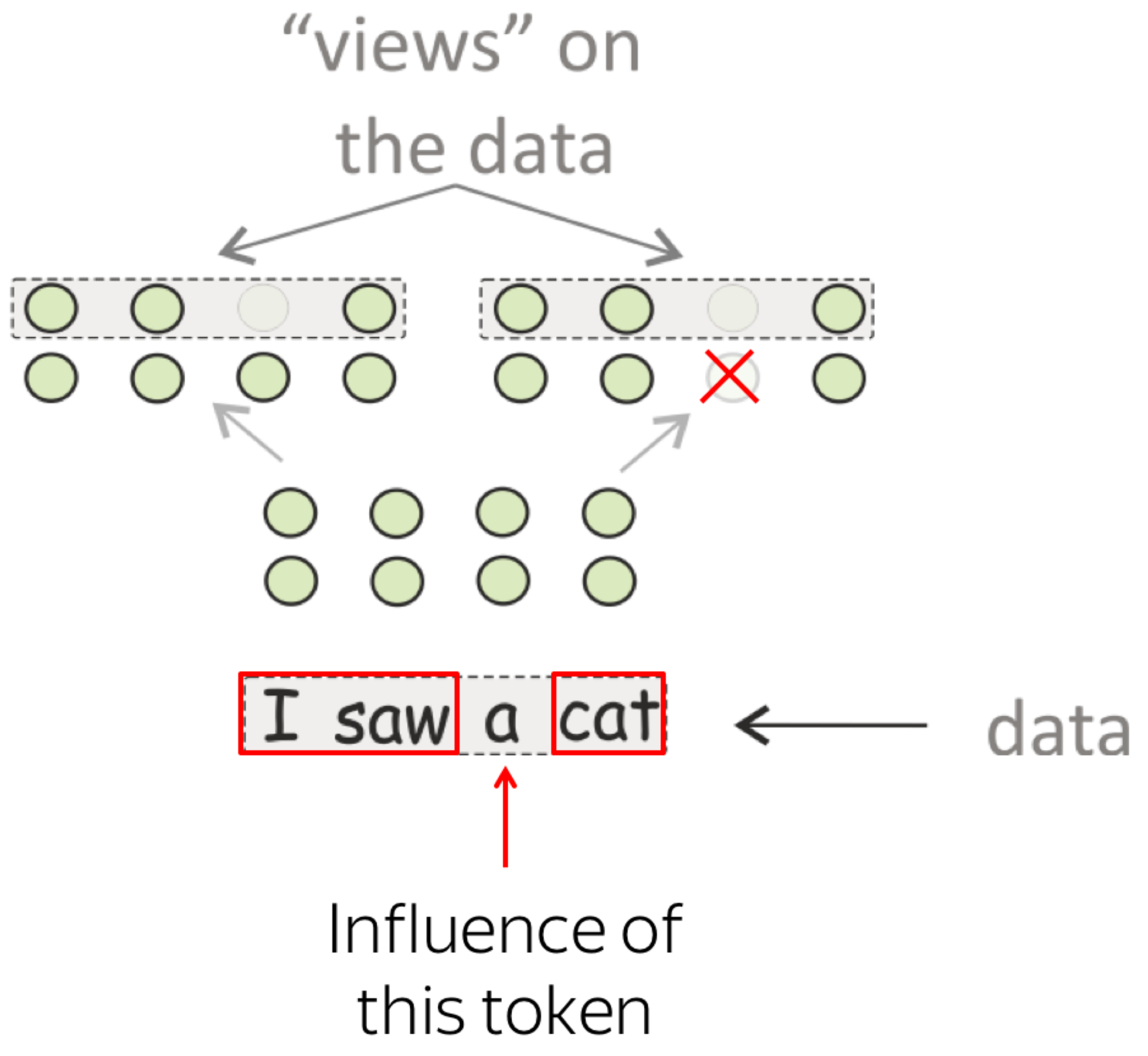


The two stages for MLM: 'context encoding' and 'token reconstruction'

Amount of change and influence

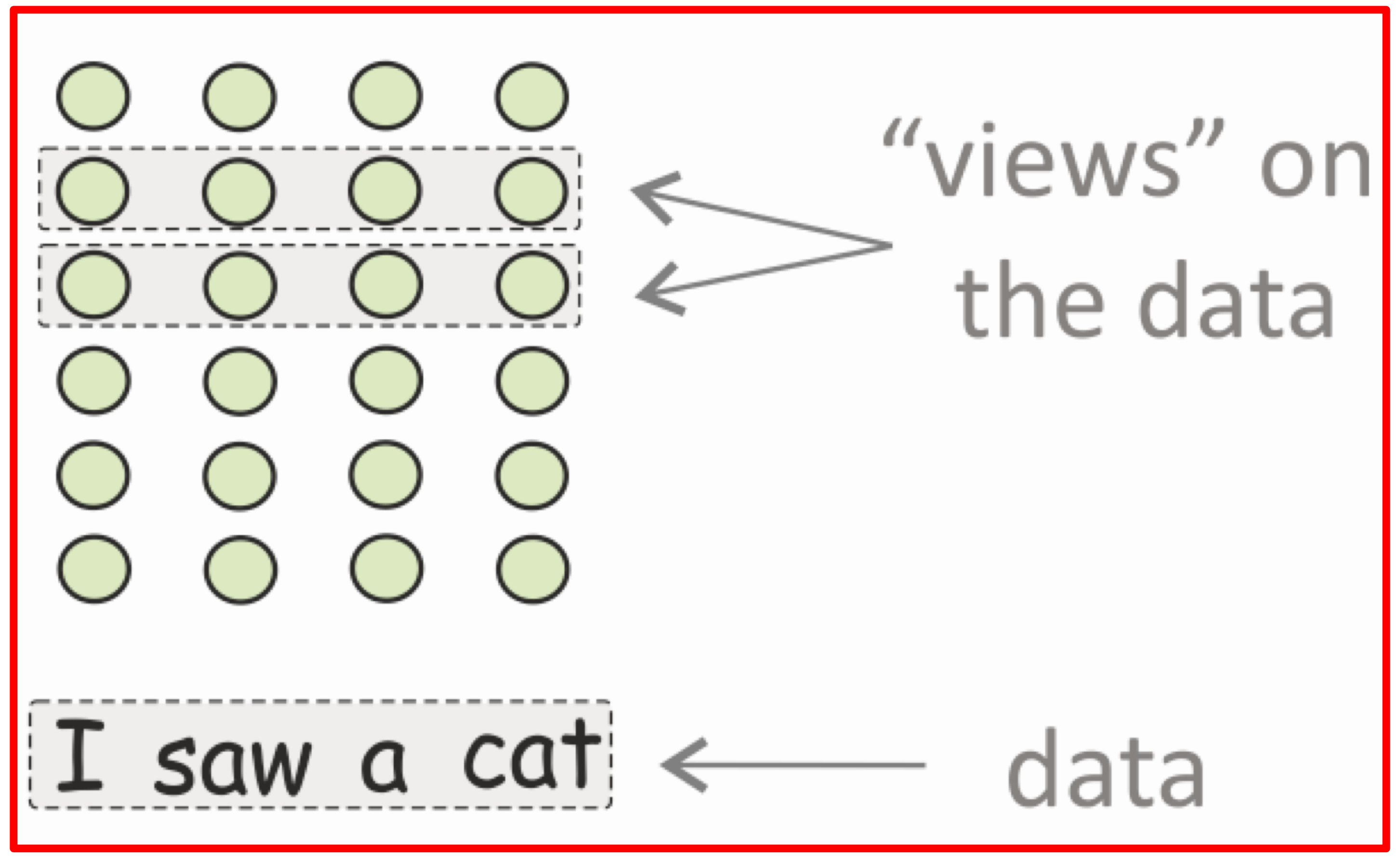
- Change: how much representations of these tokens change between layers
- Influence: how much representations of other tokens change if this token is not present



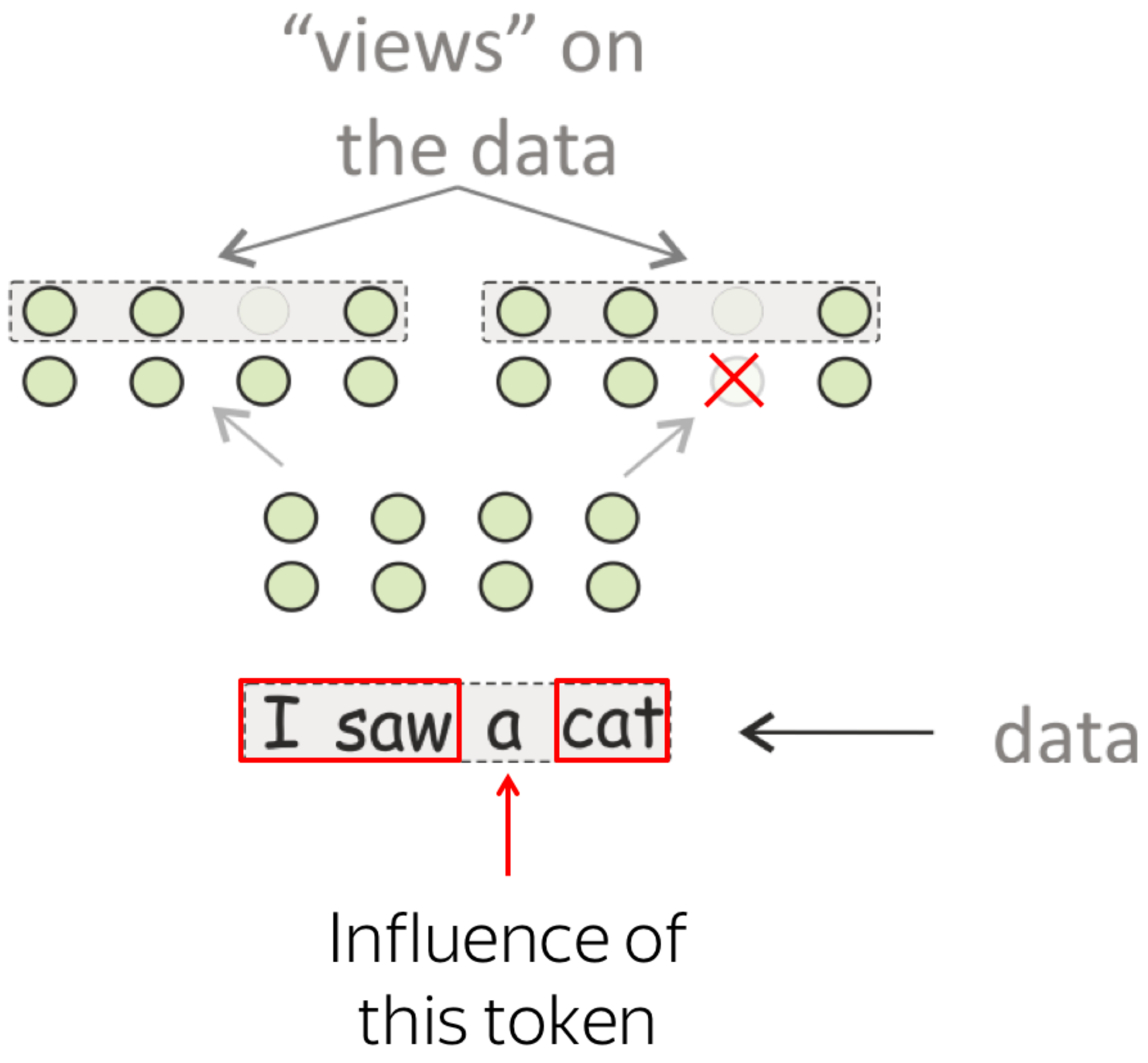


Amount of change and influence

- Change: how much representations of these tokens change between layers

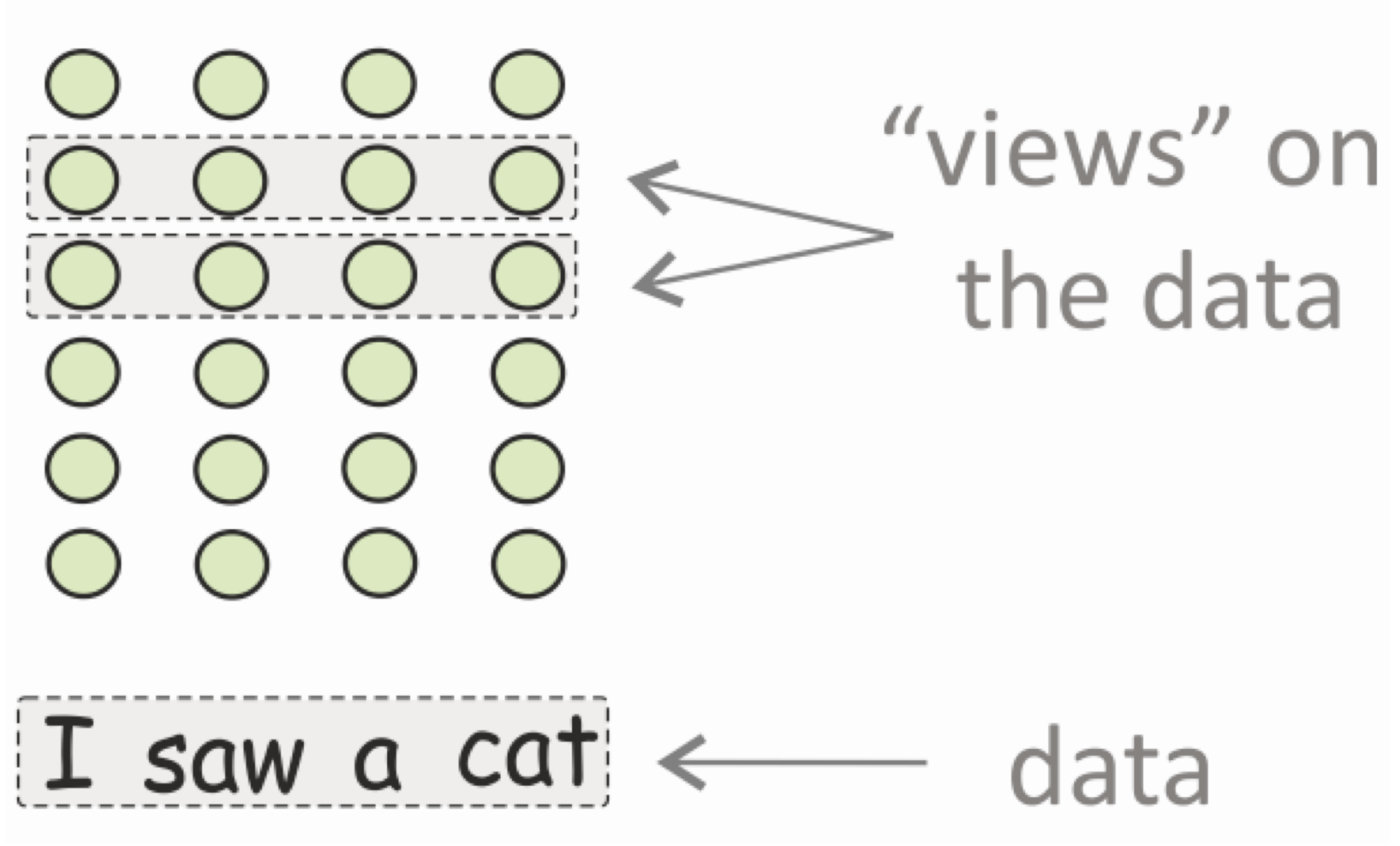


- Influence: how much representations of other tokens change if this token is not present

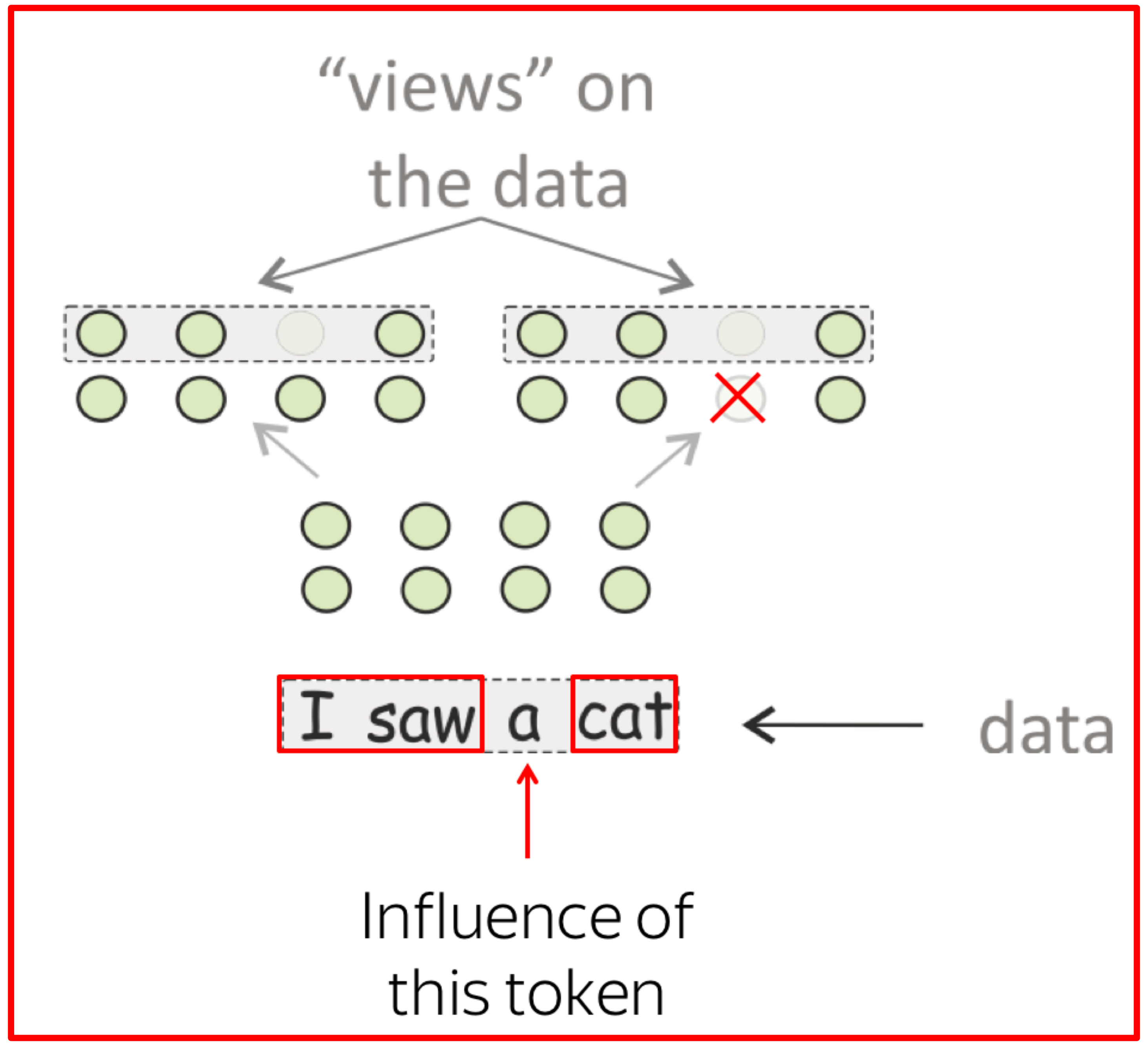


Amount of change and influence

- Change: how much representations of these tokens change between layers

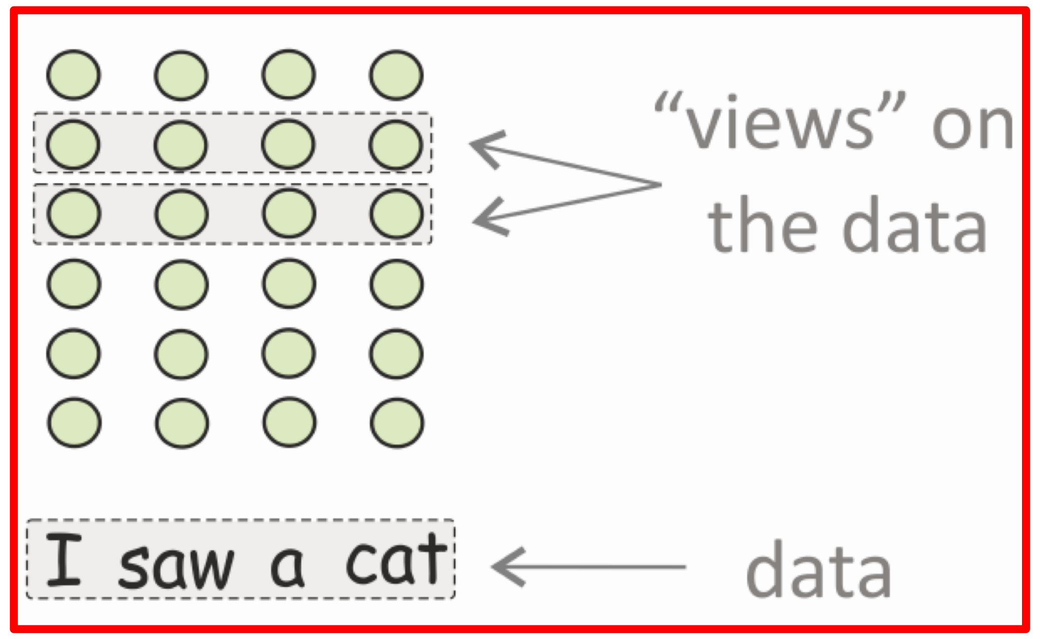


- Influence: how much representations of other tokens change if this token is not present



Varying token frequency: Amount of change

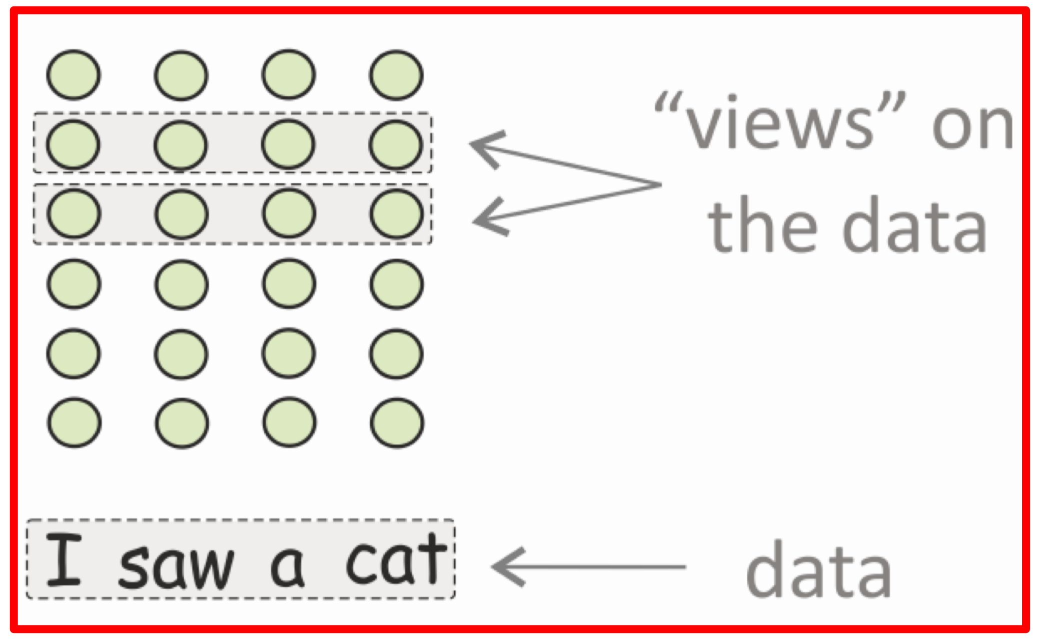
- Change: how much representations of <u>these</u> tokens change between layers



MT

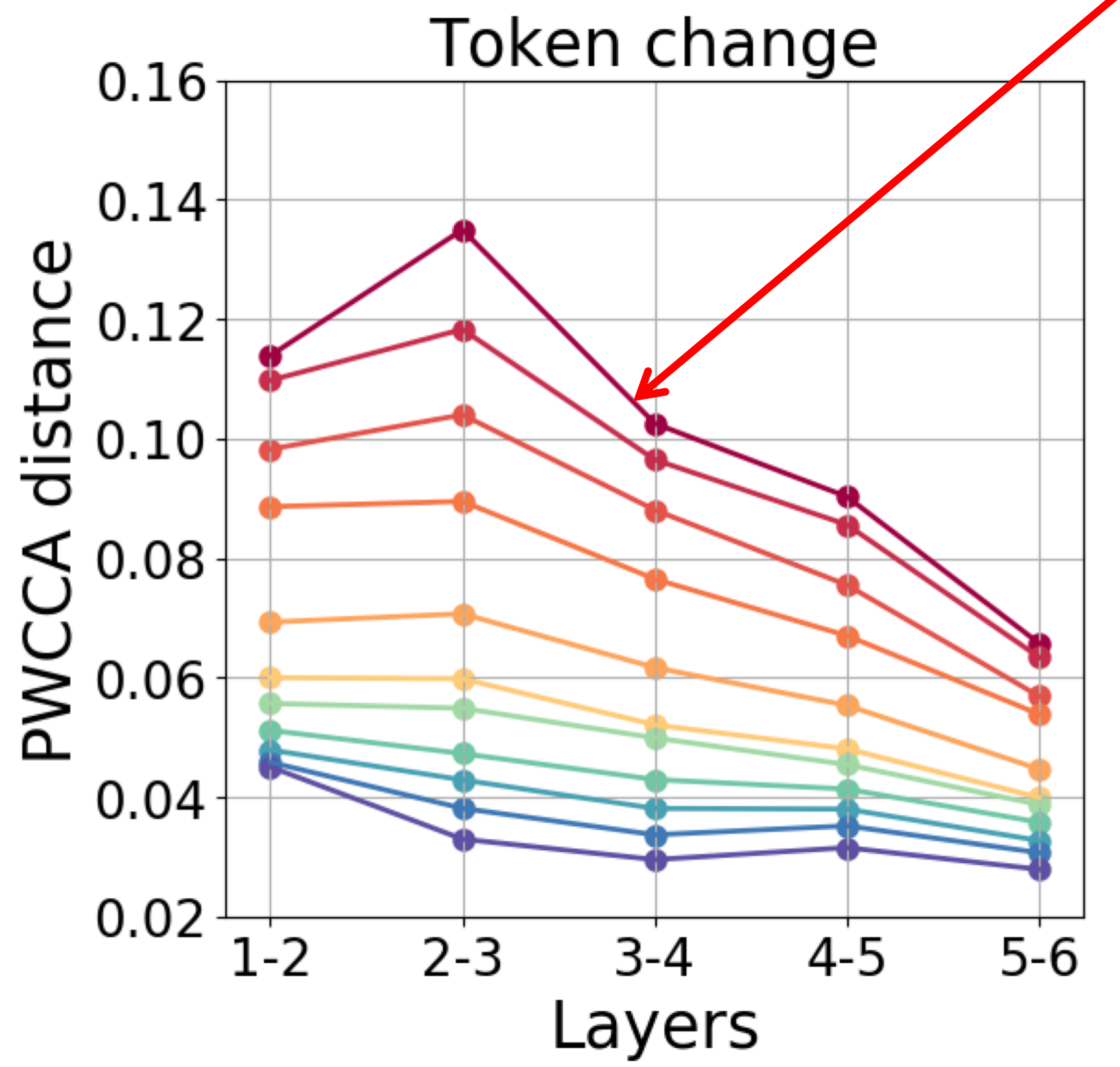
Varying token frequency: Amount of change

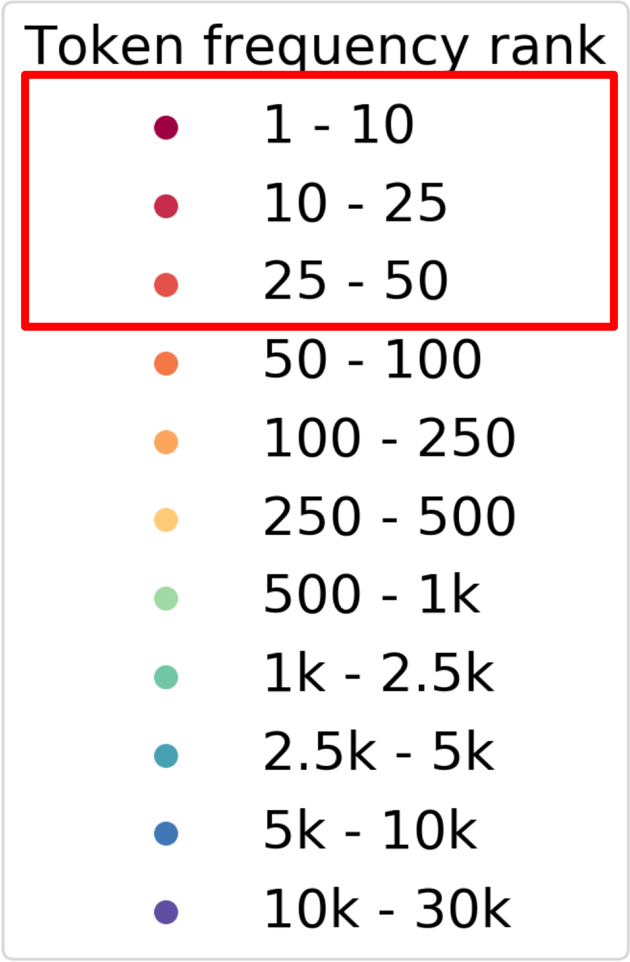
- Change: how much representations of these tokens change between layers



Frequent tokens change more than rare

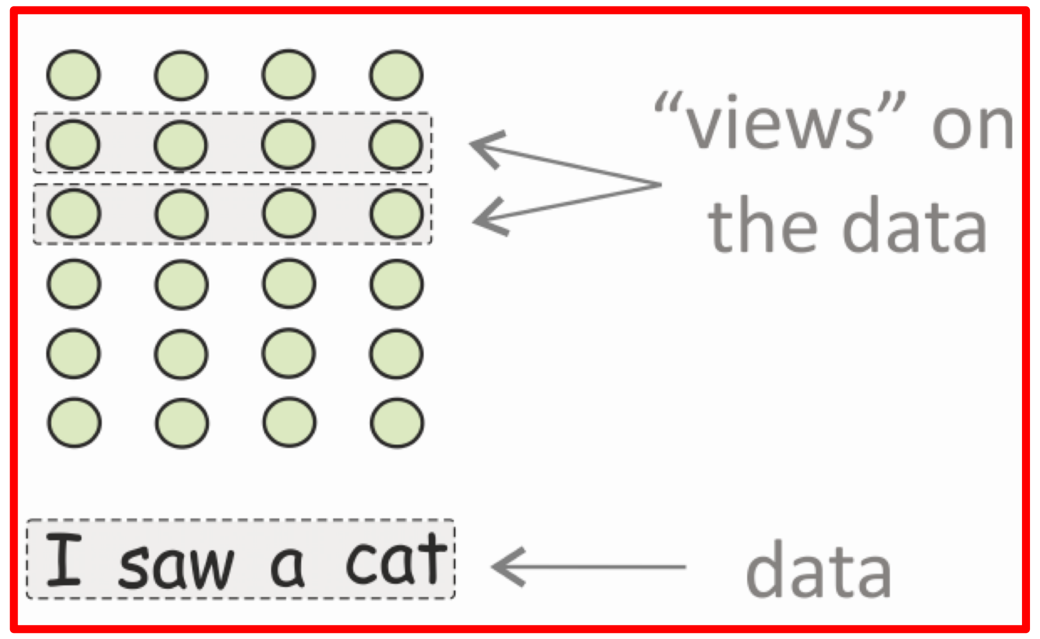
MT



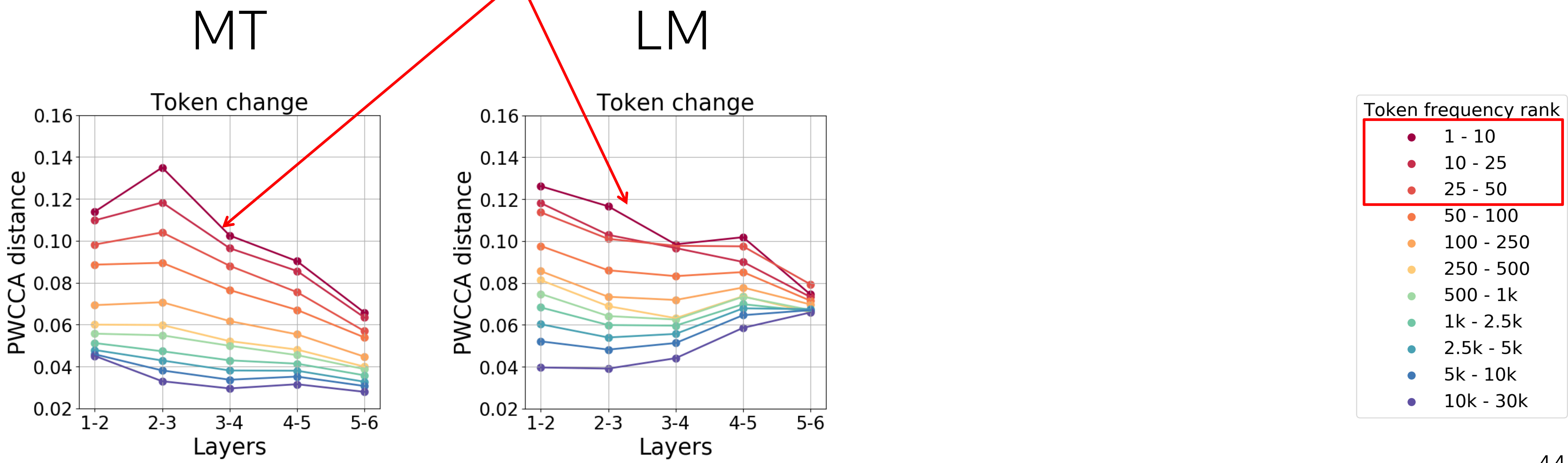


Varying token frequency: Amount of change

- Change: how much representations of these tokens change between layers

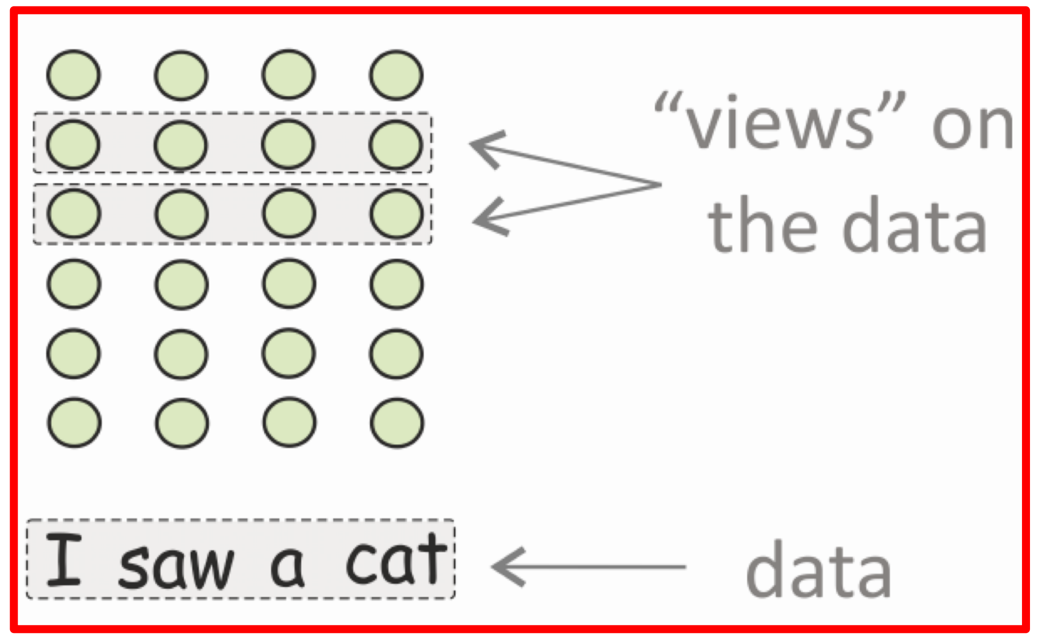


Frequent tokens change more than rare



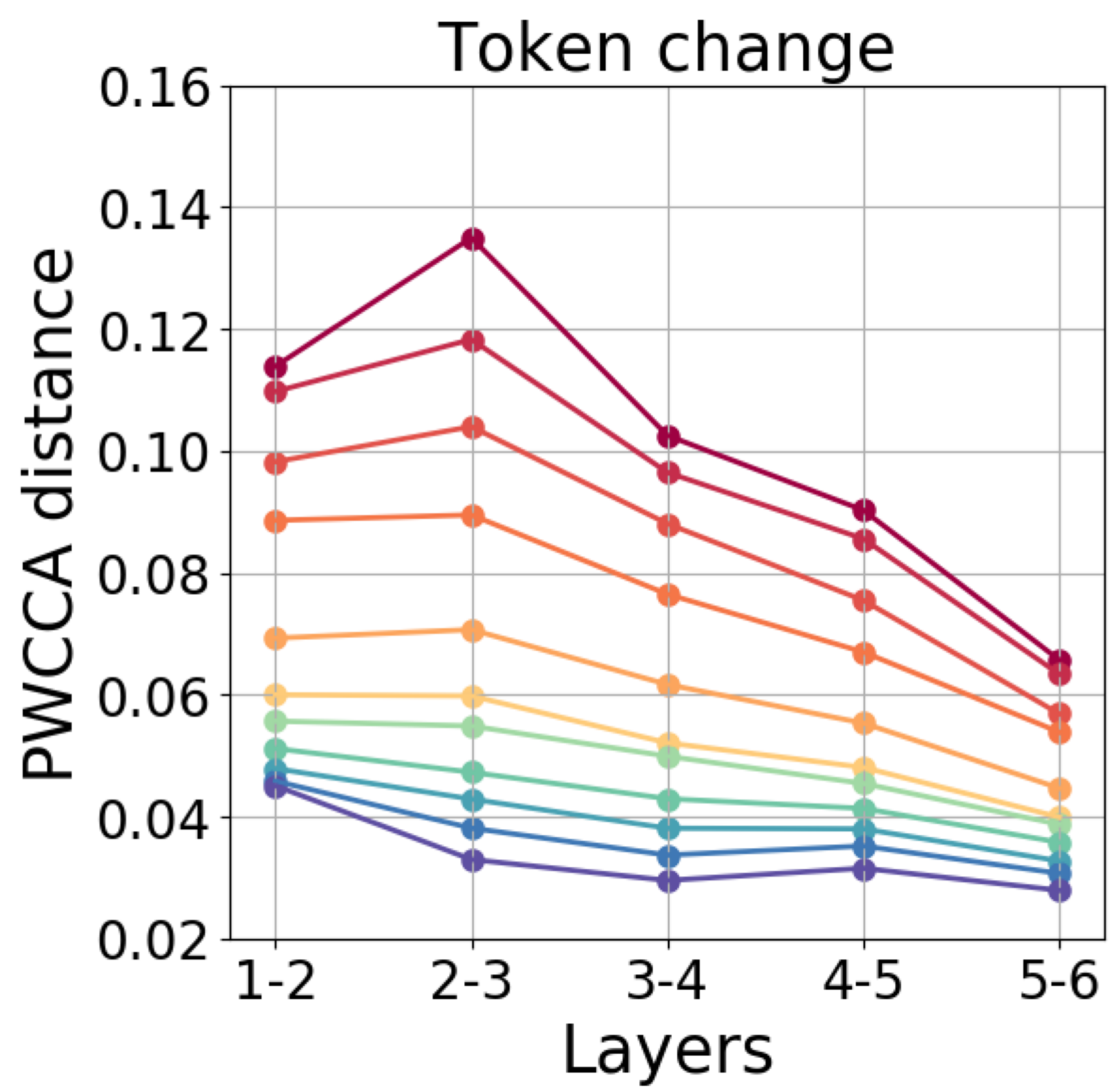
Varying token frequency: Amount of change

- Change: how much representations of these tokens change between layers

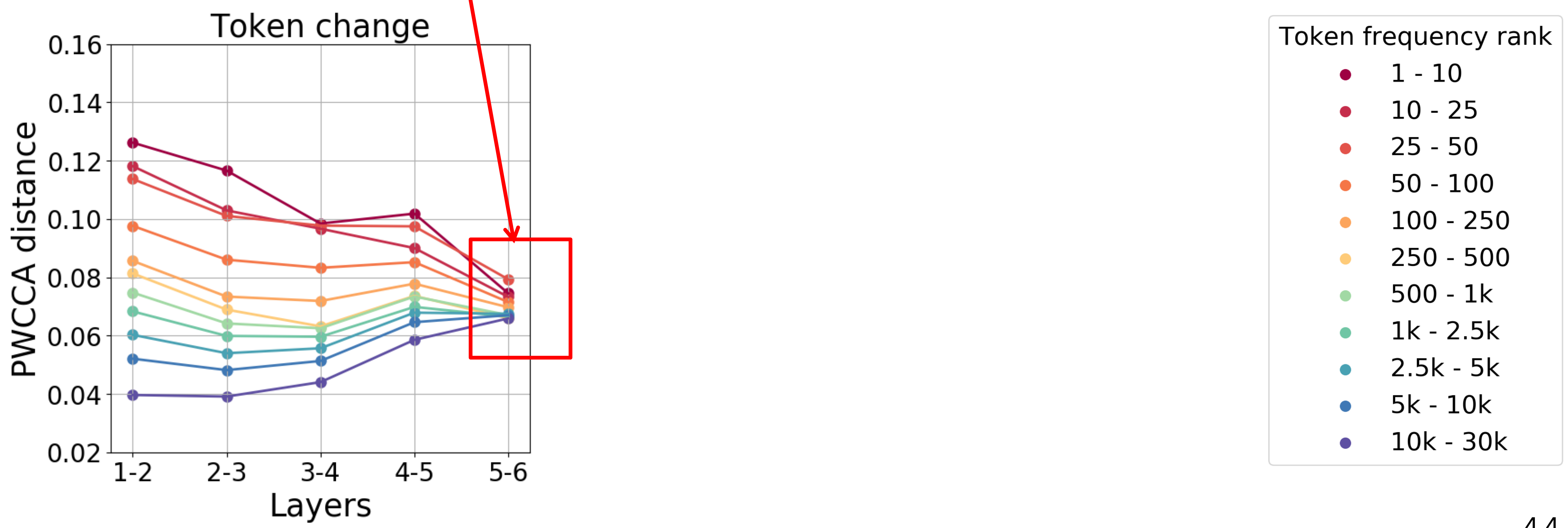


Roughly the same amount of change

MT

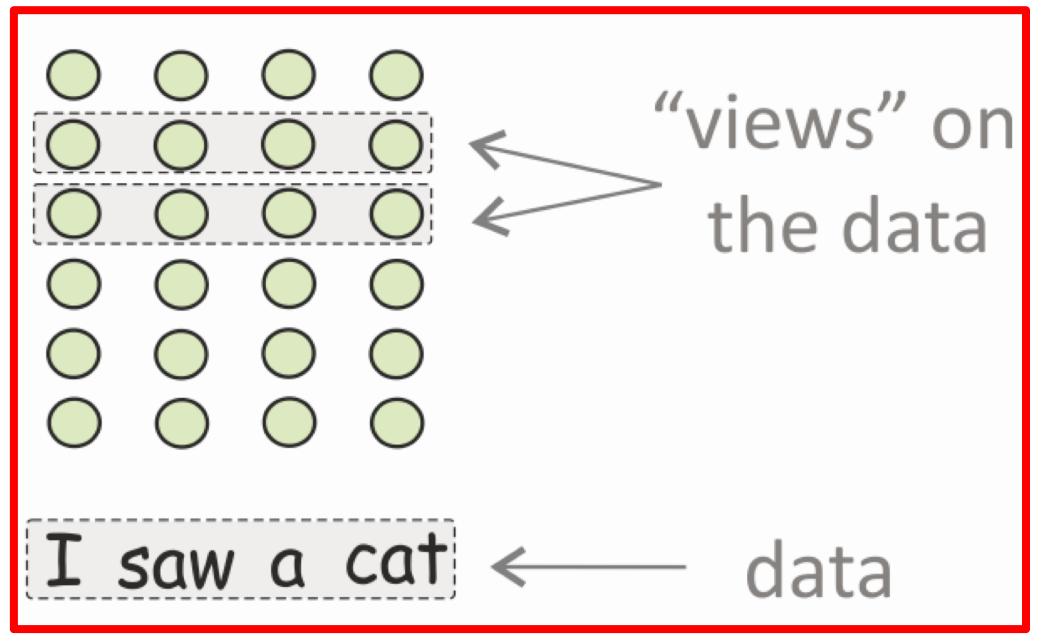


LM



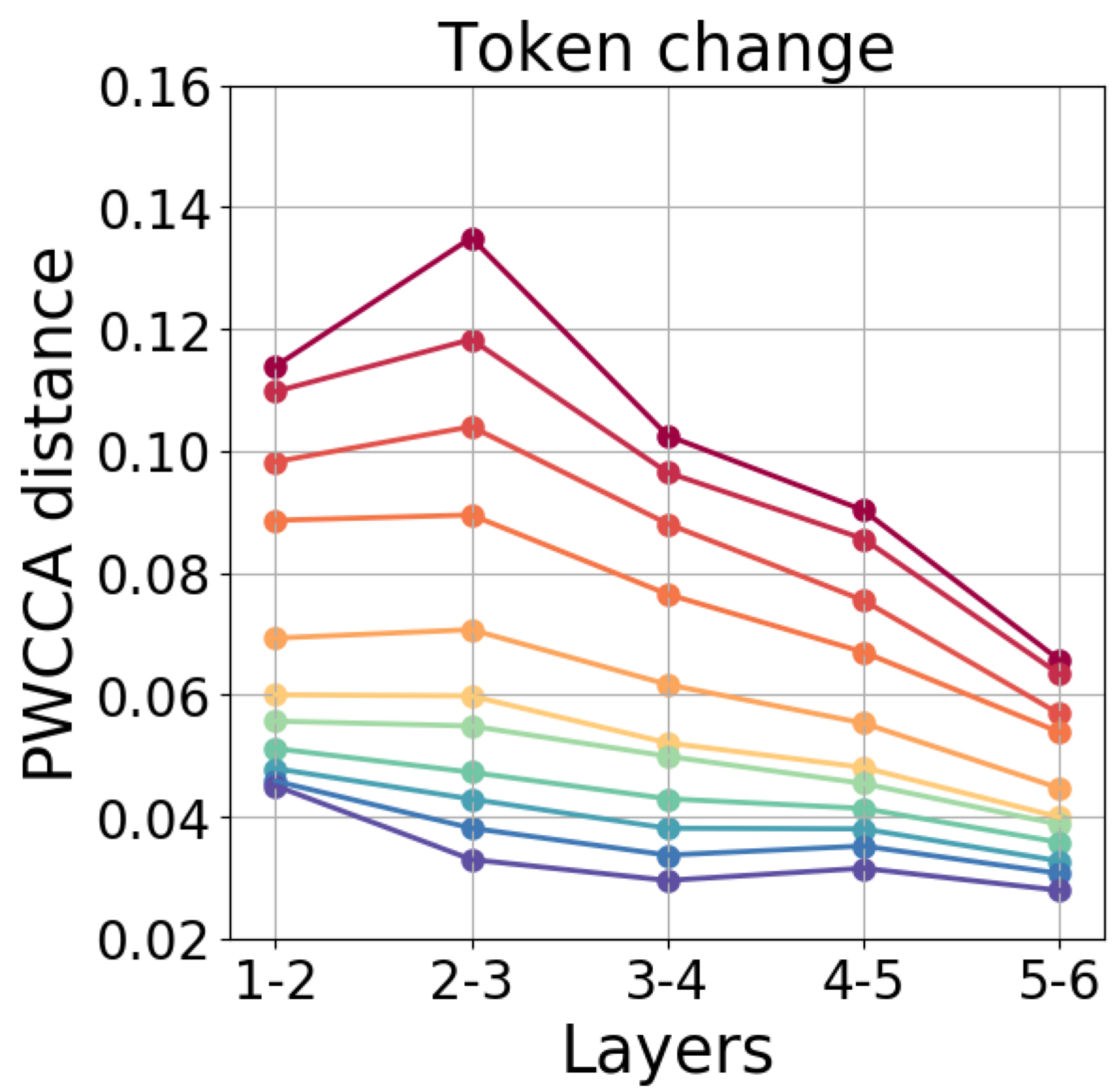
Varying token frequency: Amount of change

- Change: how much representations of these tokens change between layers

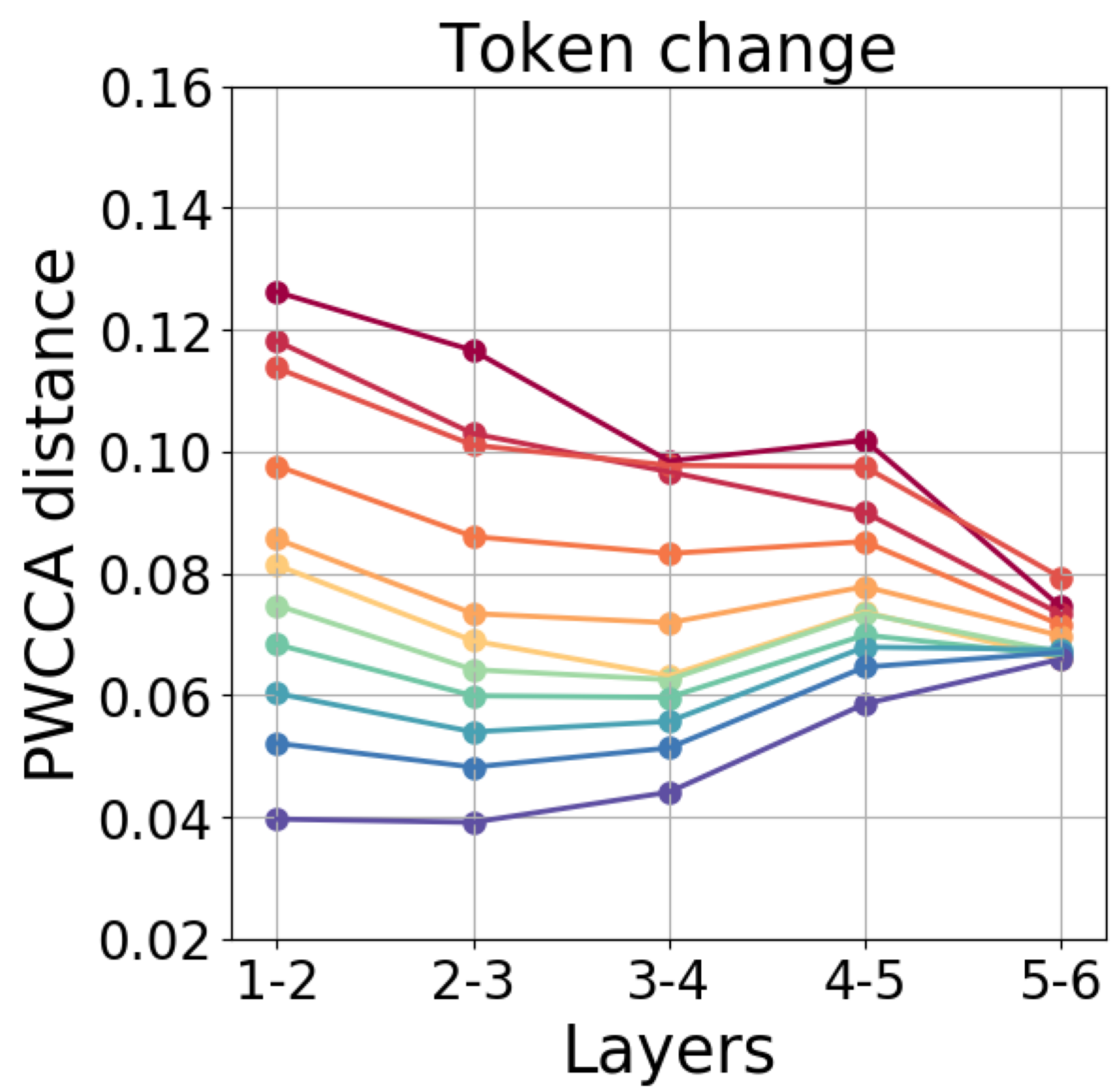


The two stages again!

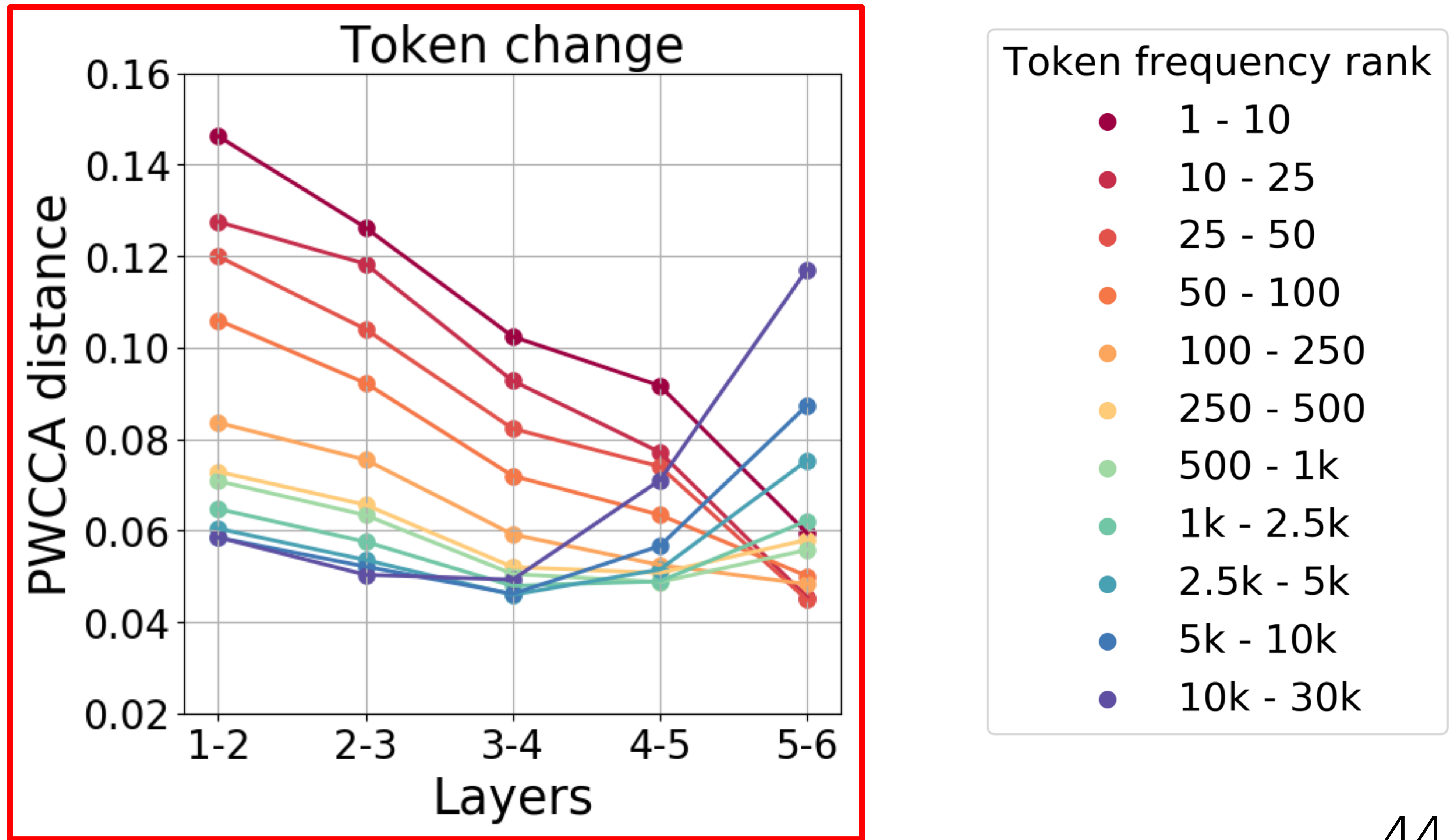
MT



LM

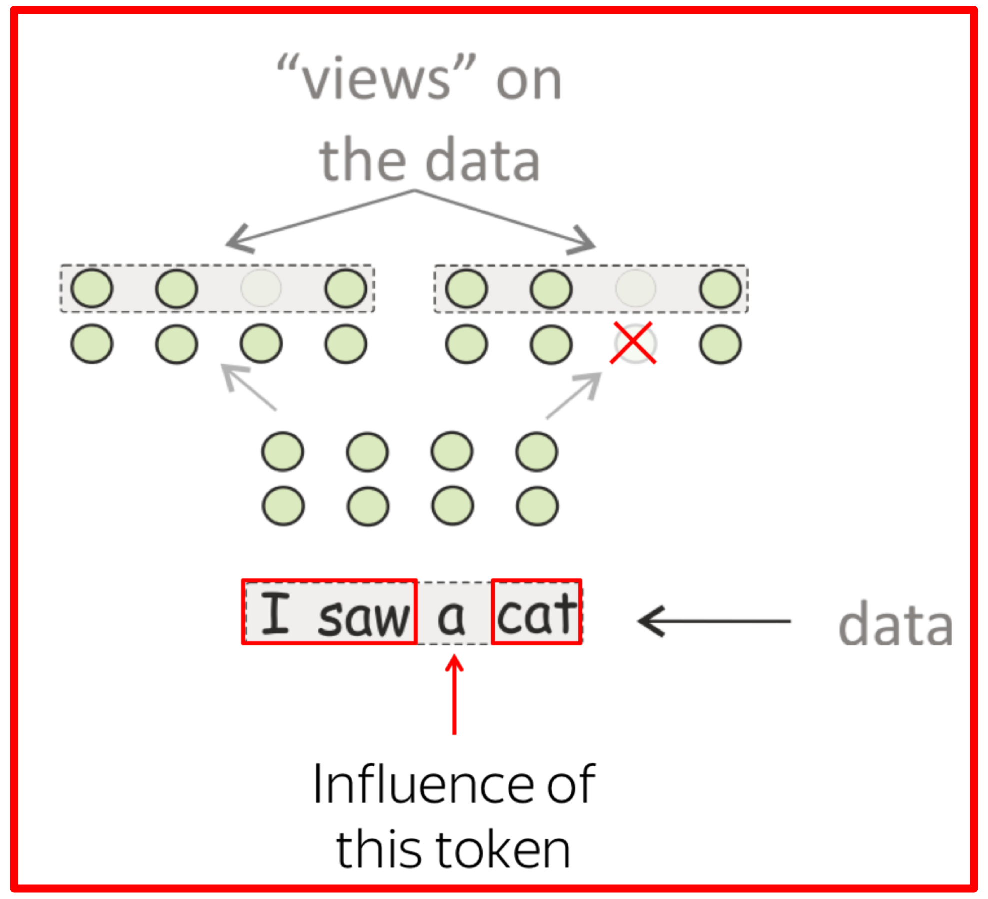


MLM



Varying token frequency: Amount of influence

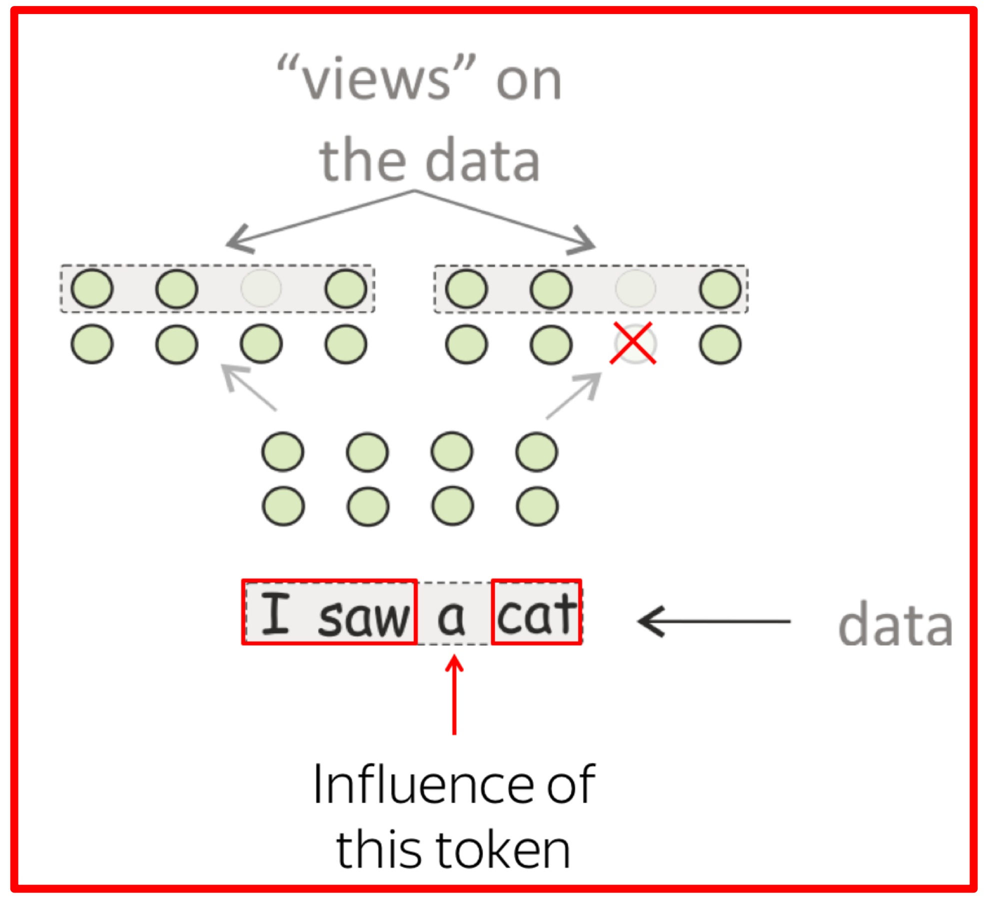
- Influence: how much representations of <u>other</u> tokens change if this token is not present



MT

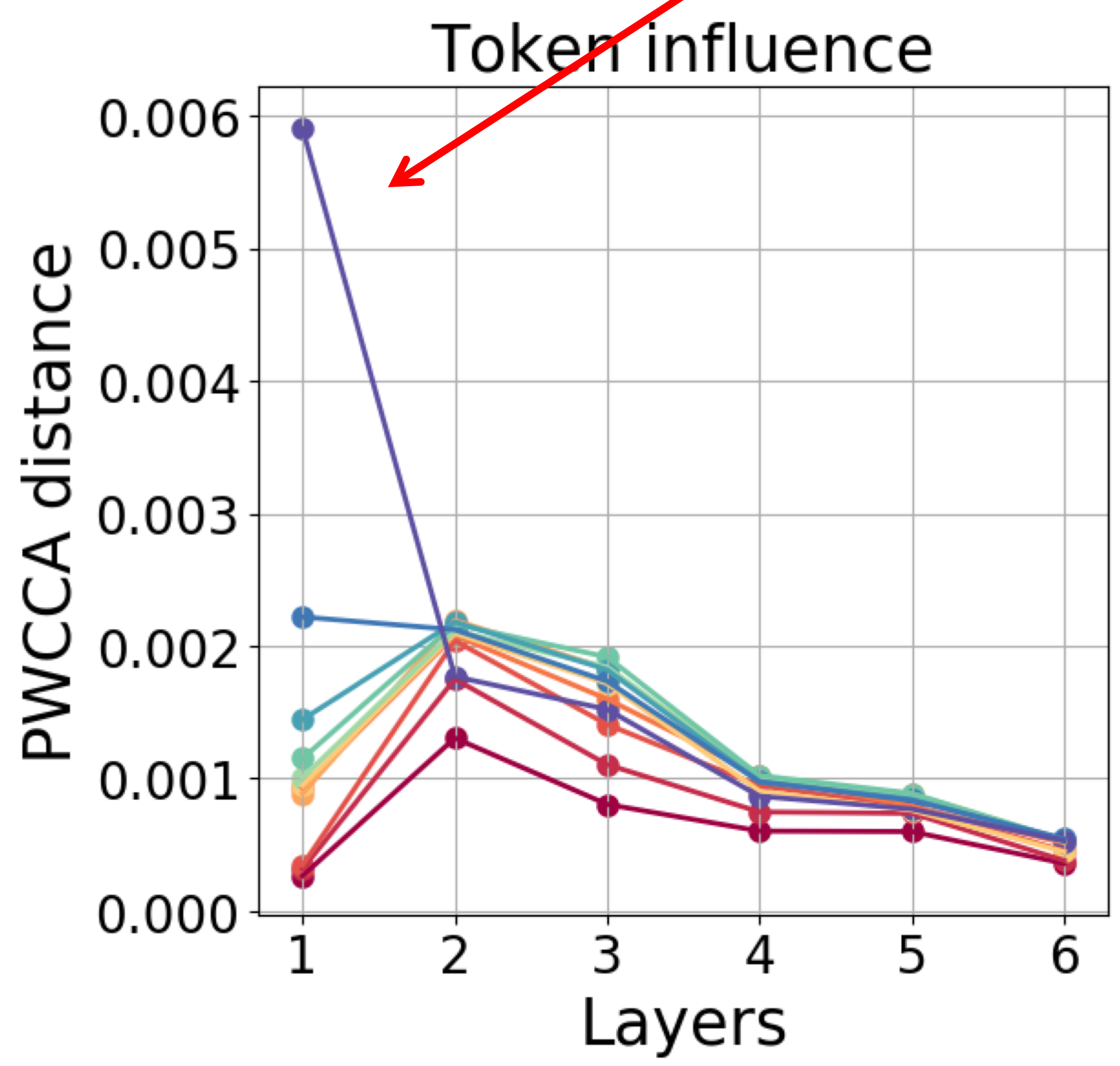
Varying token frequency: Amount of influence

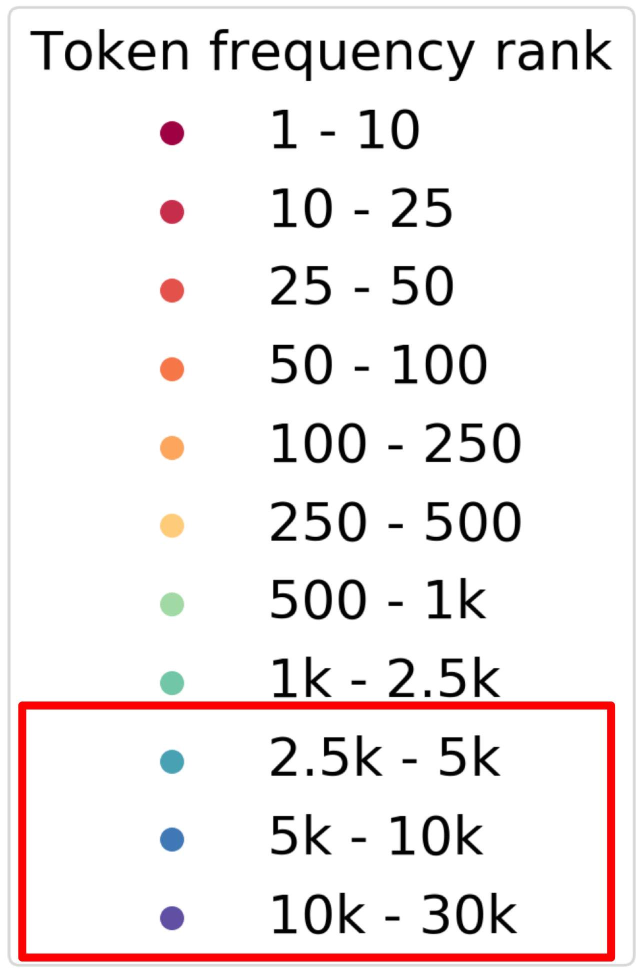
- Influence: how much representations of other tokens change if this token is not present



Rare tokens influence more

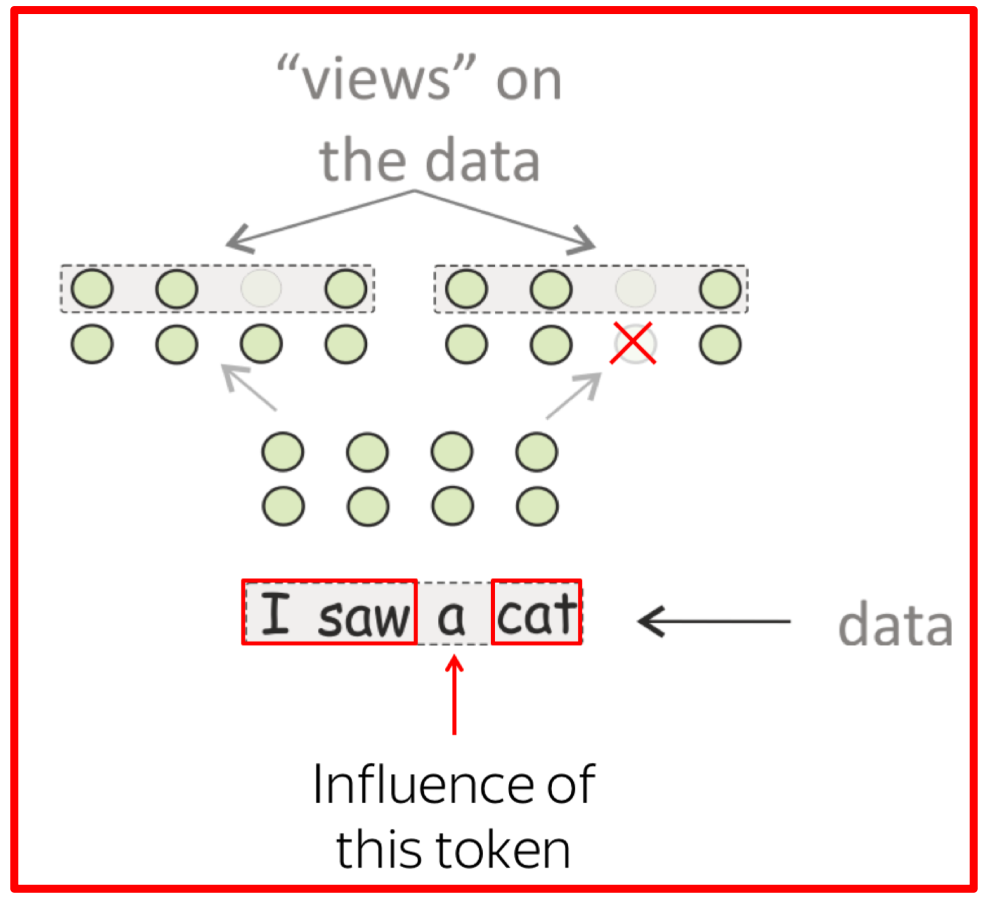
MT



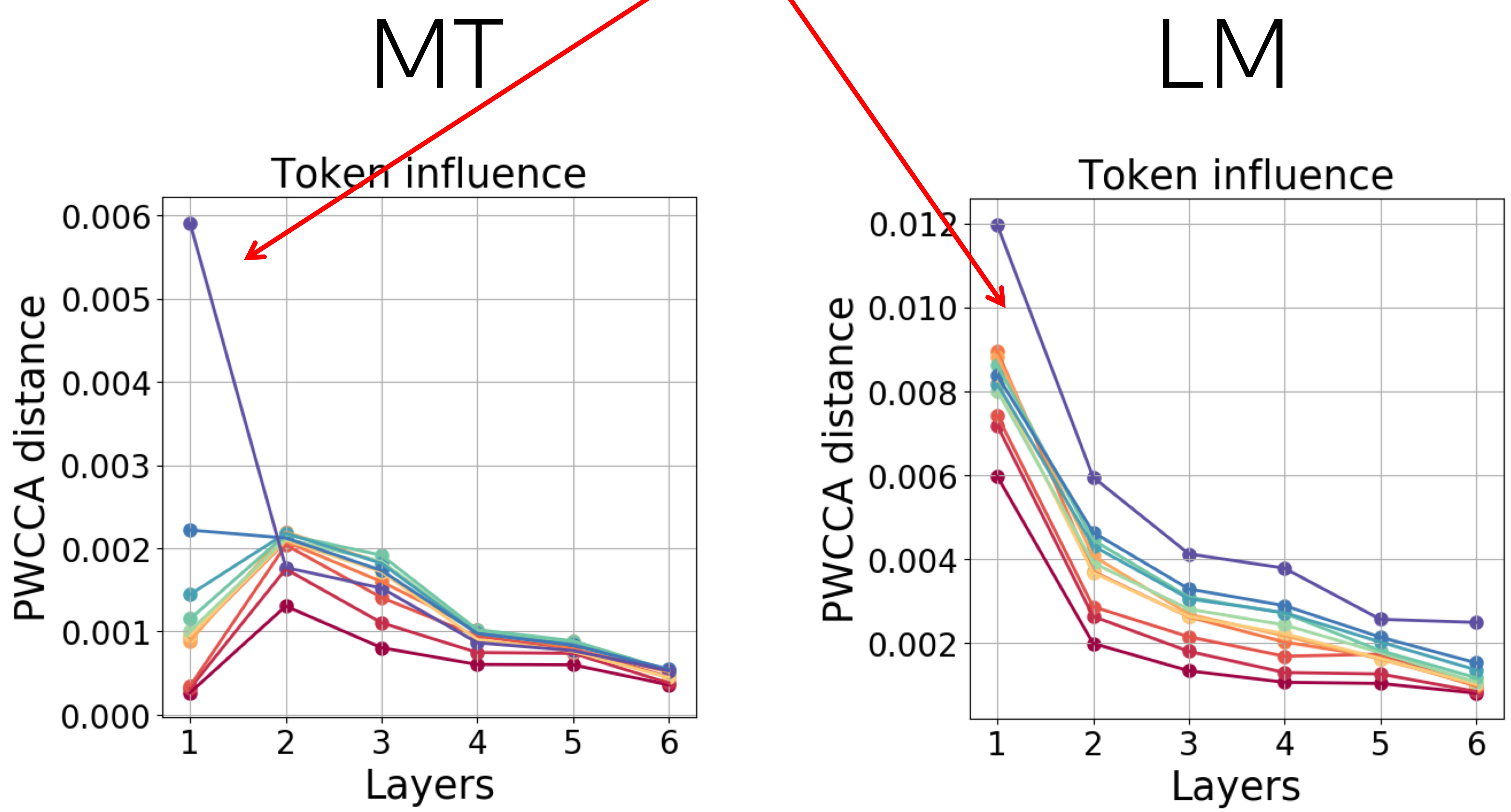


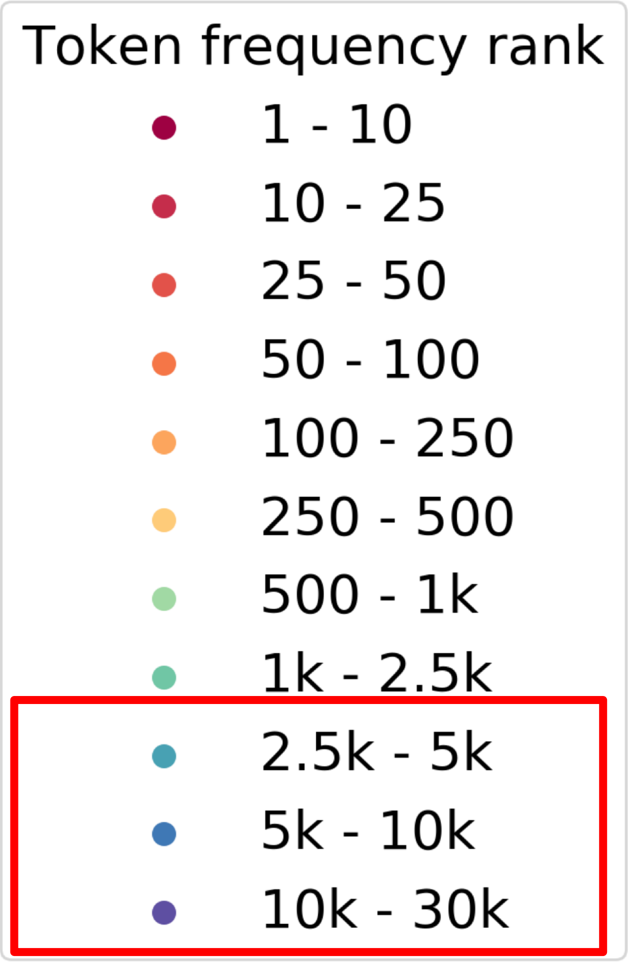
Varying token frequency: Amount of influence

- Influence: how much representations of other tokens change if this token is not present



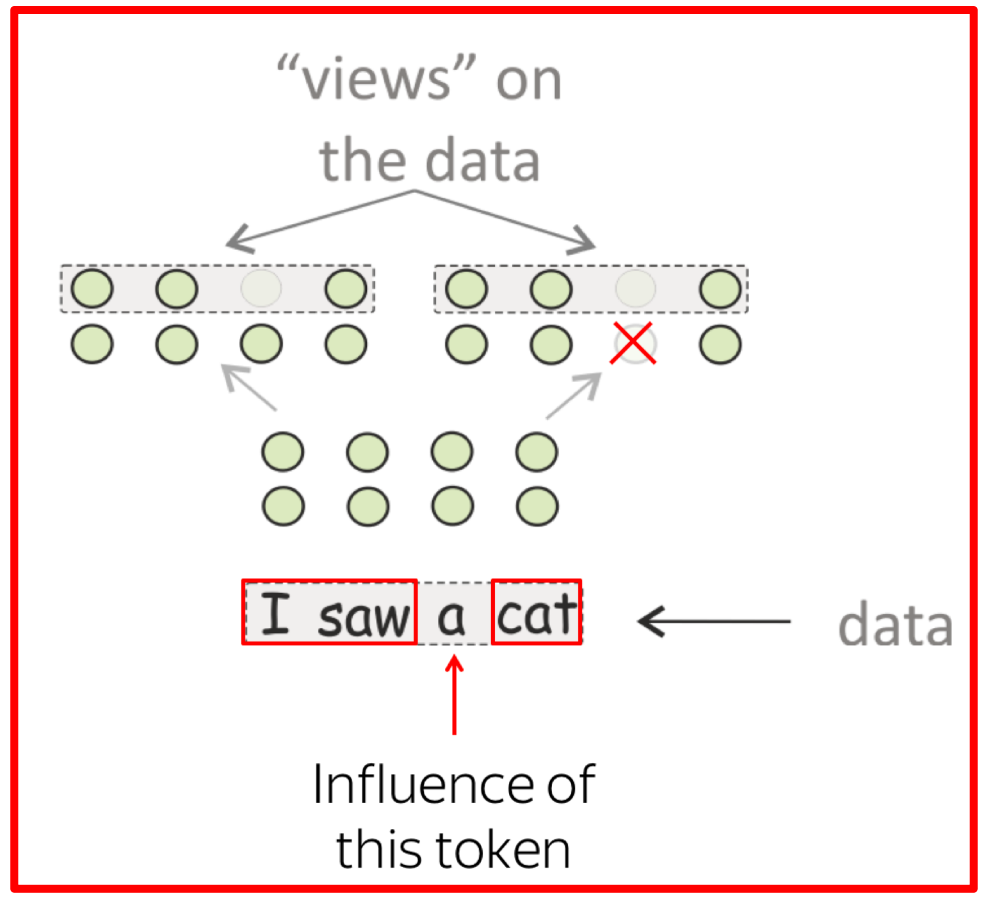
Rare tokens influence more



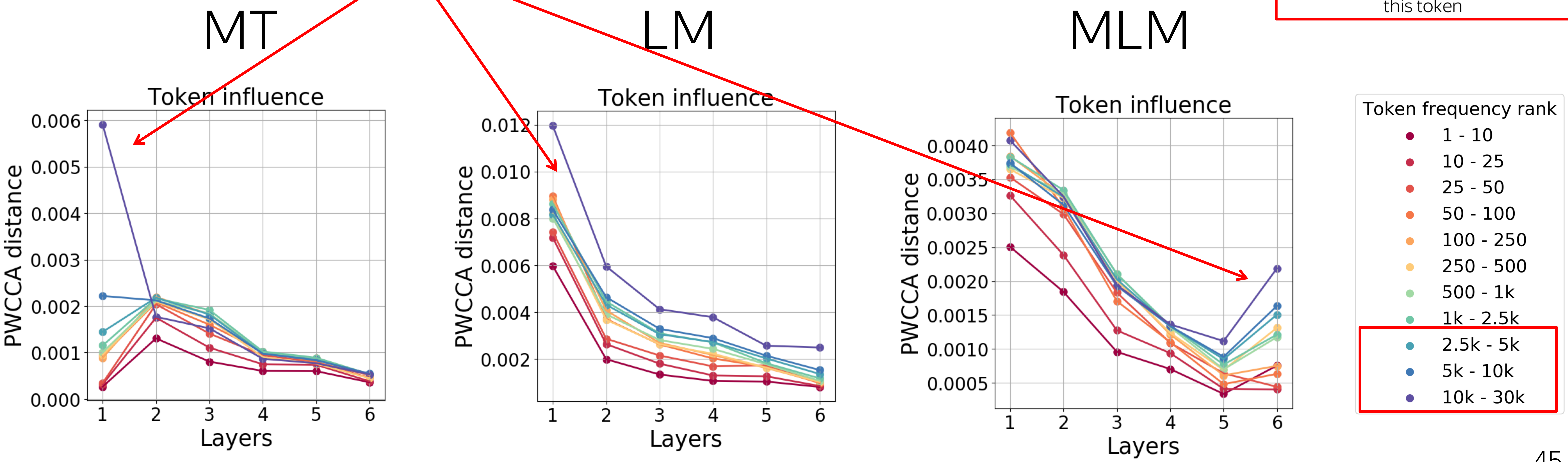


Varying token frequency: Amount of influence

- Influence: how much representations of other tokens change if this token is not present



Rare tokens influence more



Plan

- Evolution of representations of individual tokens
- Training objectives: LM, MLM, MT
- ”Puzzles” from previous work
- The Information-Bottleneck: our point of view
- Experiments
    - Information Bottleneck for token representations
    - Analyzing changes and influences
    - …

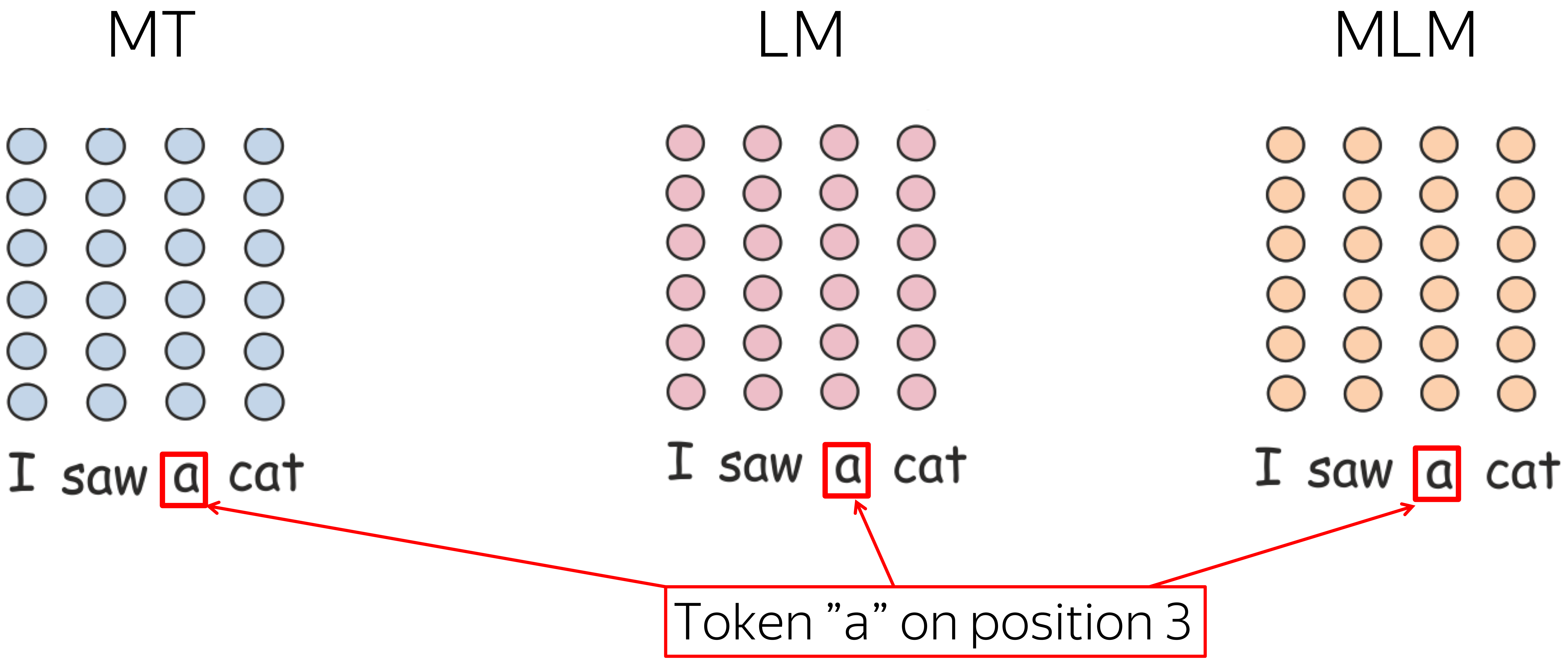
Plan

- Evolution of representations of individual tokens
- Training objectives: LM, MLM, MT
- ”Puzzles” from previous work
- The Information-Bottleneck: our point of view
- Experiments
    - Information Bottleneck for token representations
    - Analyzing changes and influences
    - What does a layer represent?

What does a layer represent?

The bottom-up evolution

- All models start from the same representation: token identity and position



Preserving token identity

The cats **are** tired of sitting on a mat

The cats **are** hungry

This **is** a great opportunity

Are you happy?

It **is** raining

This mat **is** full of cats

Simon $\boxed{\text{is}}$ a lazy cat

Is it Jane?

What **is** an evolution?

These apples **are** so tasty!

They **were** on vacation last week

Was it a good vacation?

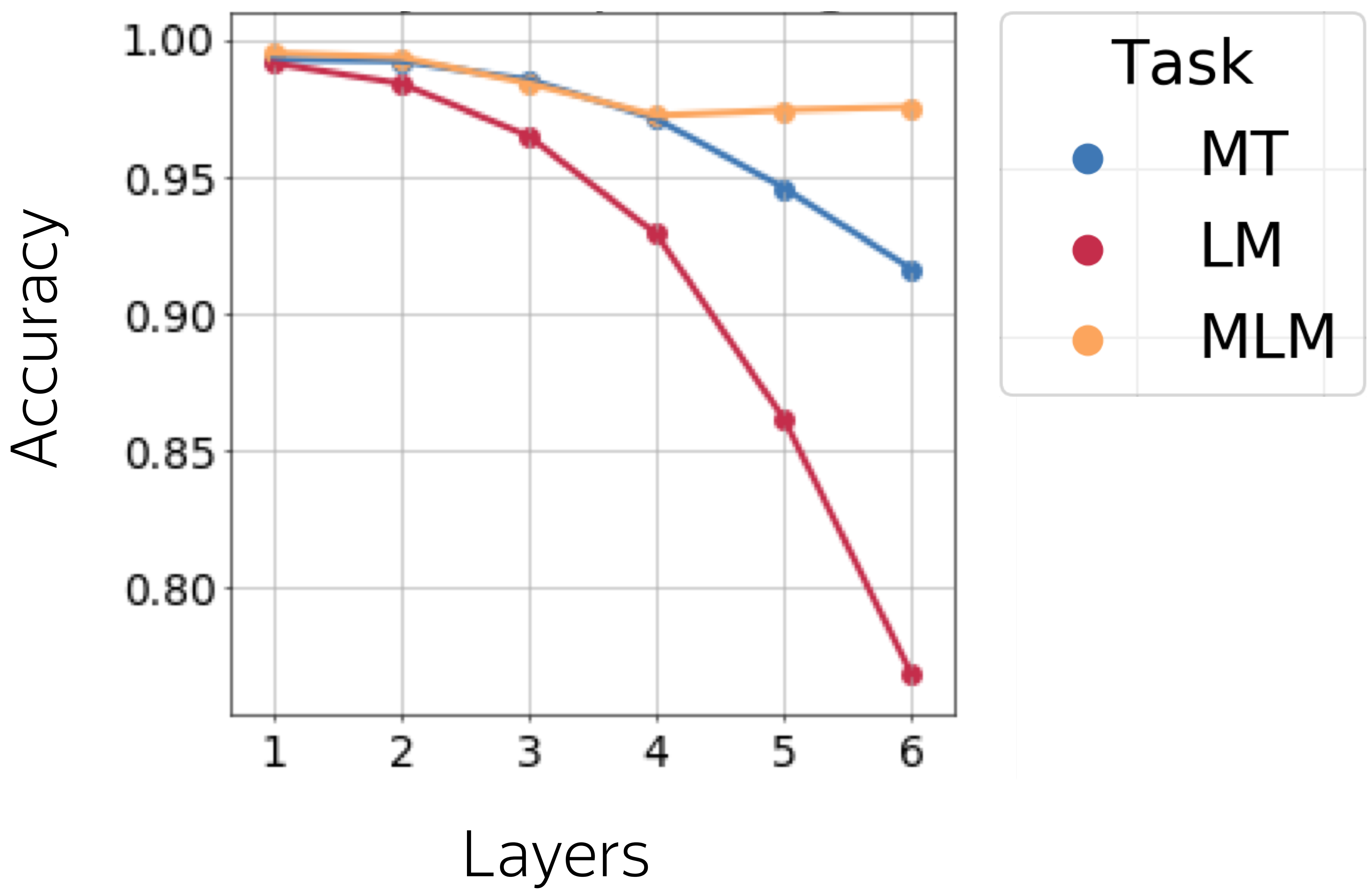
I **was** glad to see you

- Take large number of representations of different tokens

Preserving token identity

- Take large number of representations of different tokens
- Evaluate the proportion of top-k neighbors which have the same token identity

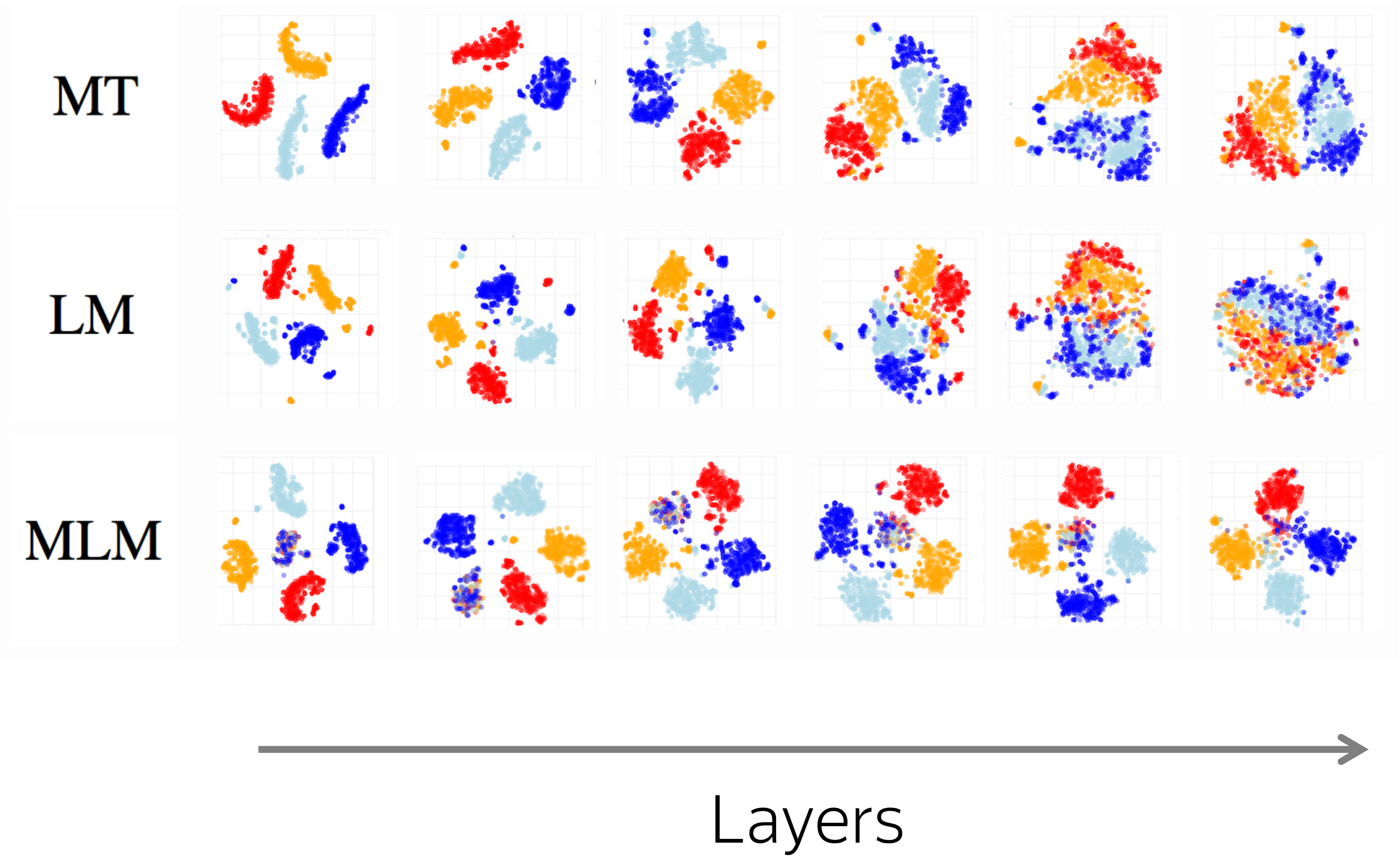
Preserving token identity



Really similar to the MI results!

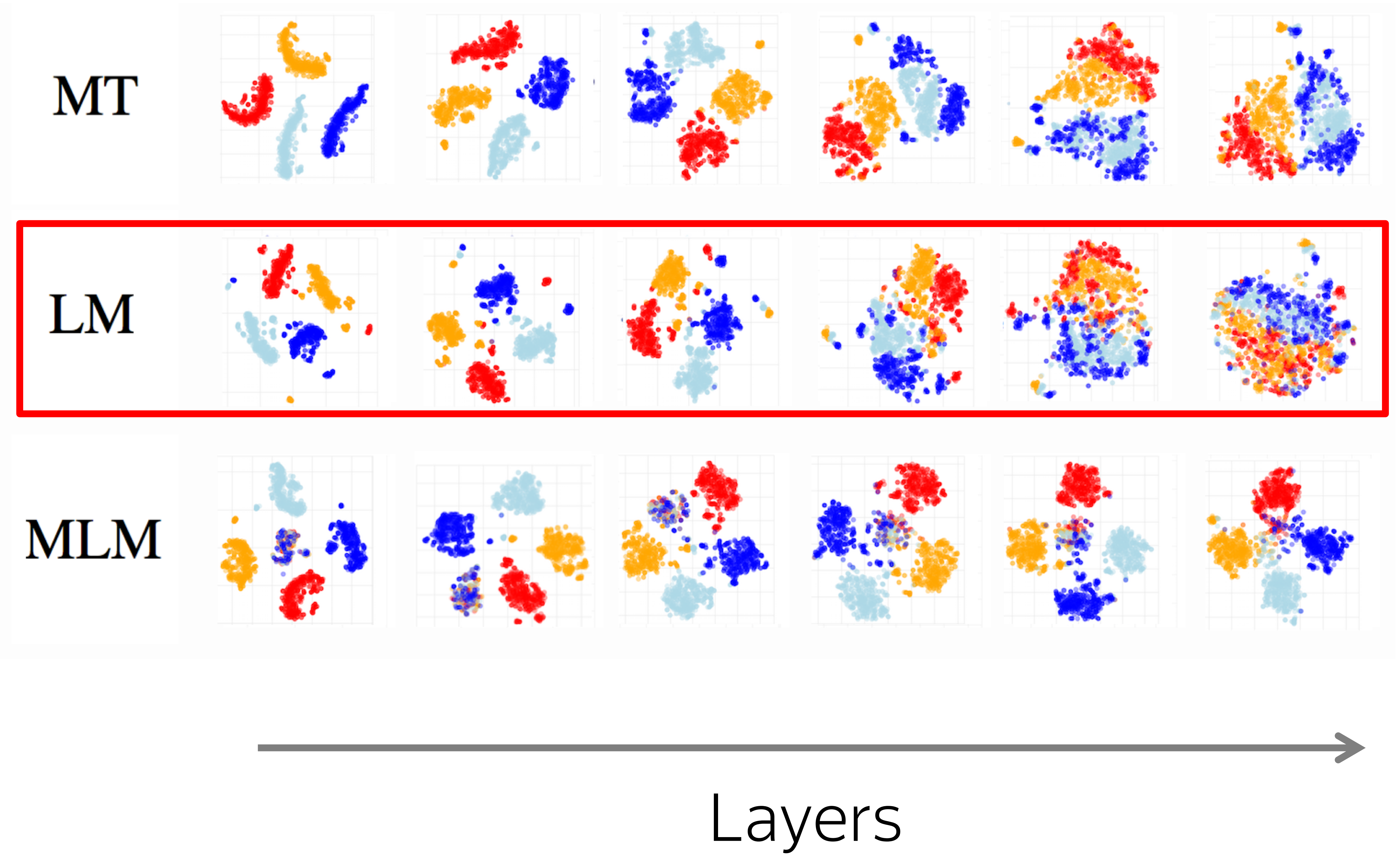
Preserving token identity

- t-SNE of different occurrences of the tokens is, are, was, were



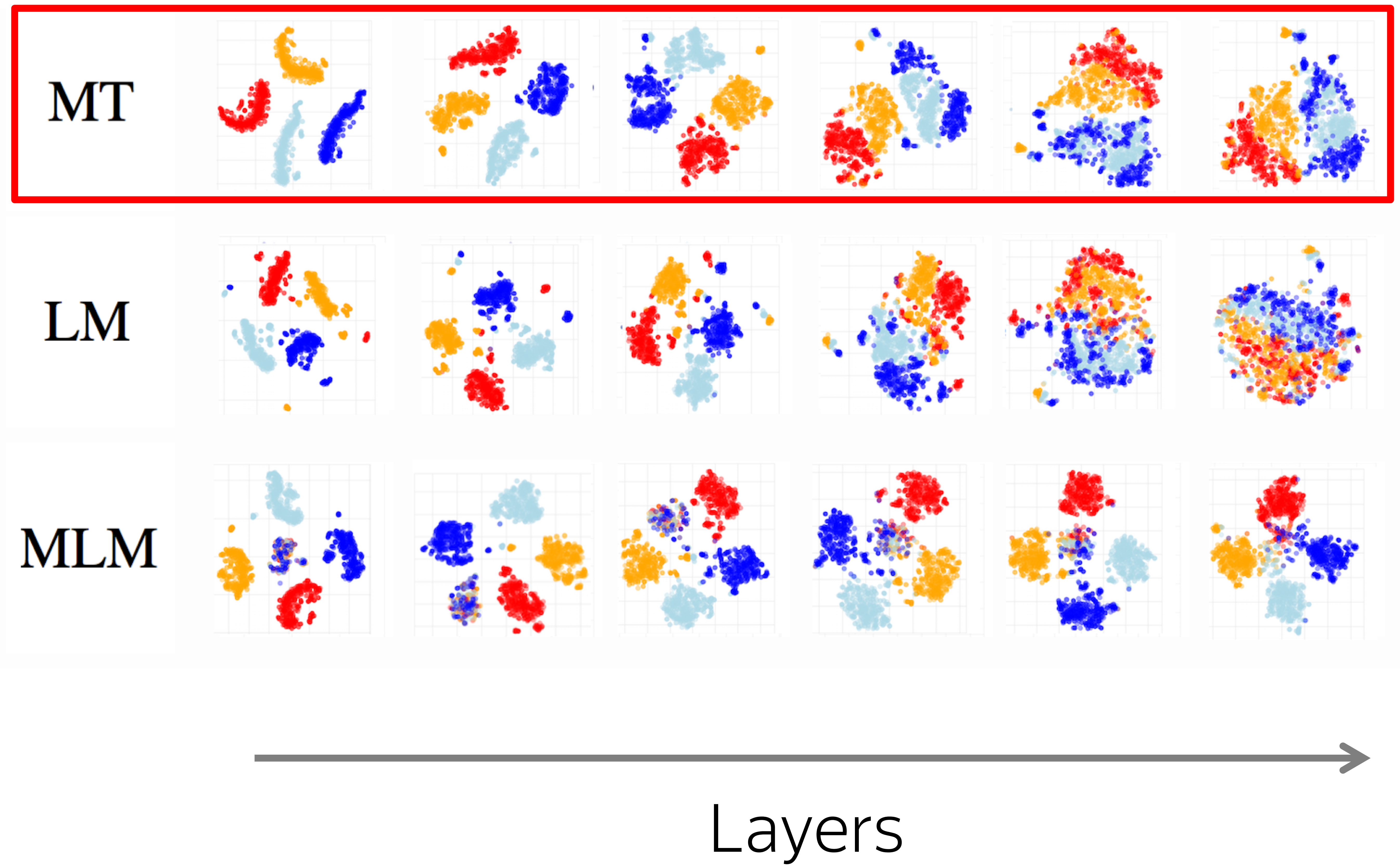
Preserving token identity

- t-SNE of different occurrences of the tokens is, are, was, were



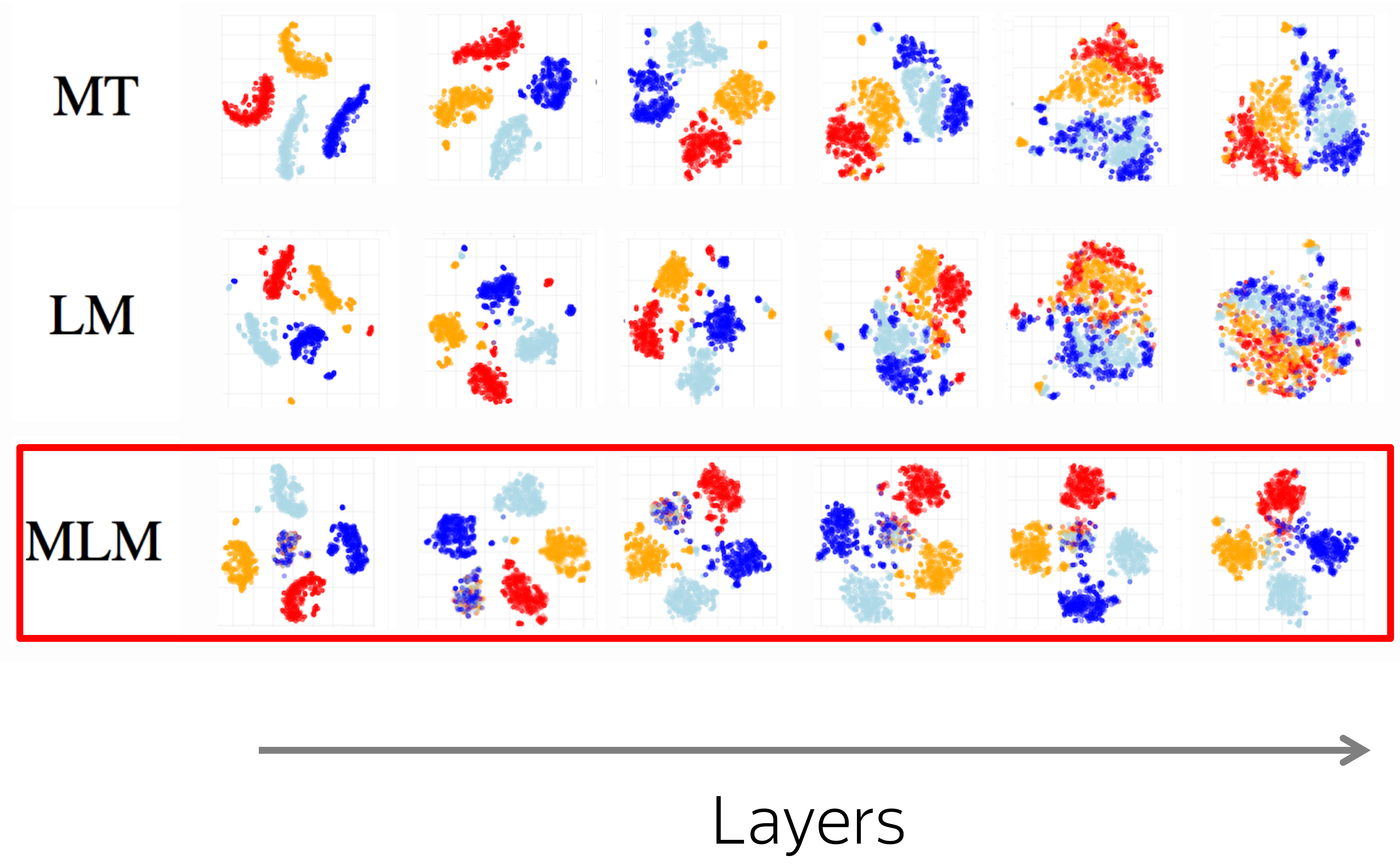
Preserving token identity

- t-SNE of different occurrences of the tokens is, are, was, were



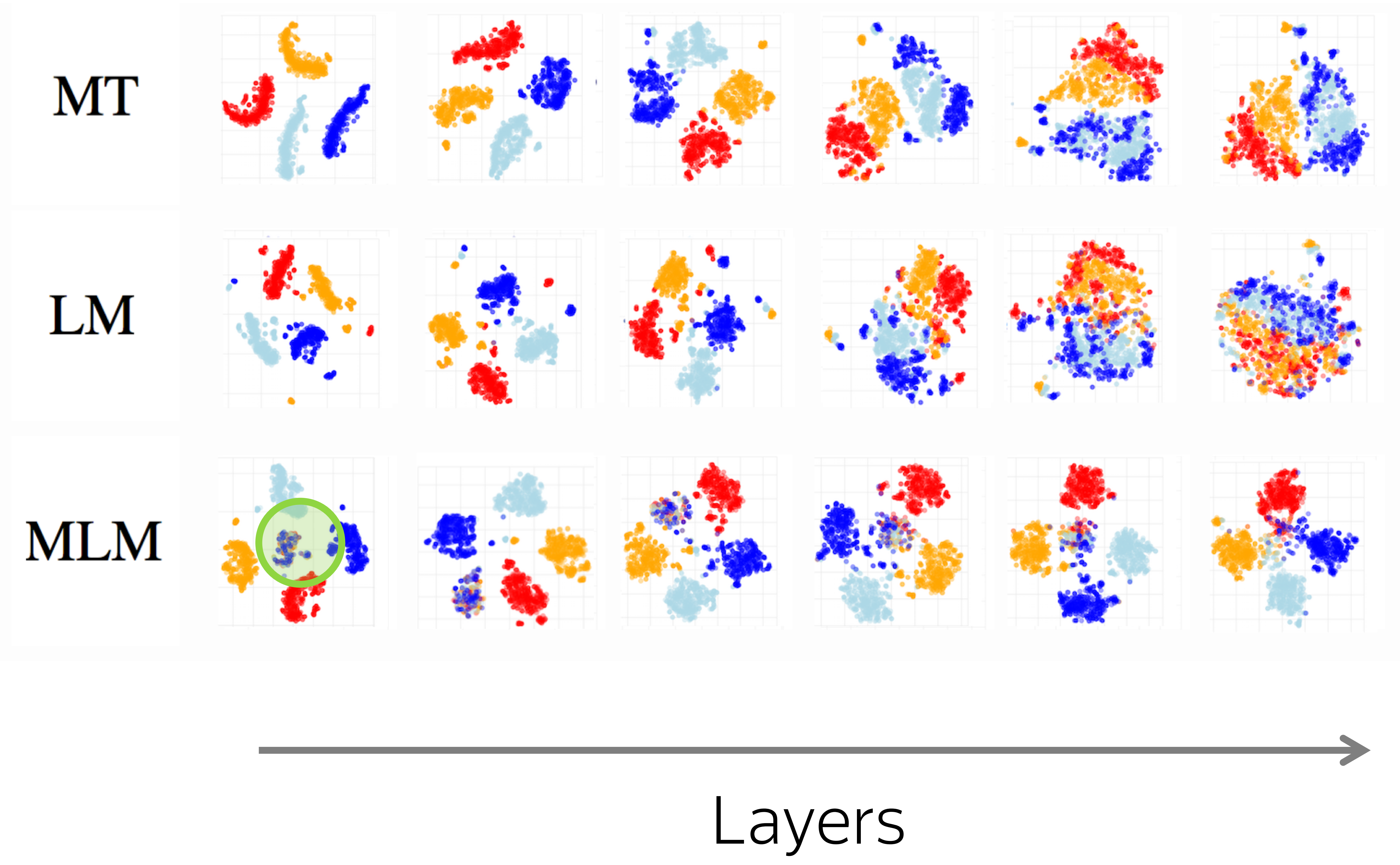
Preserving token identity

- t-SNE of different occurrences of the tokens is, are, was, were



Preserving token identity

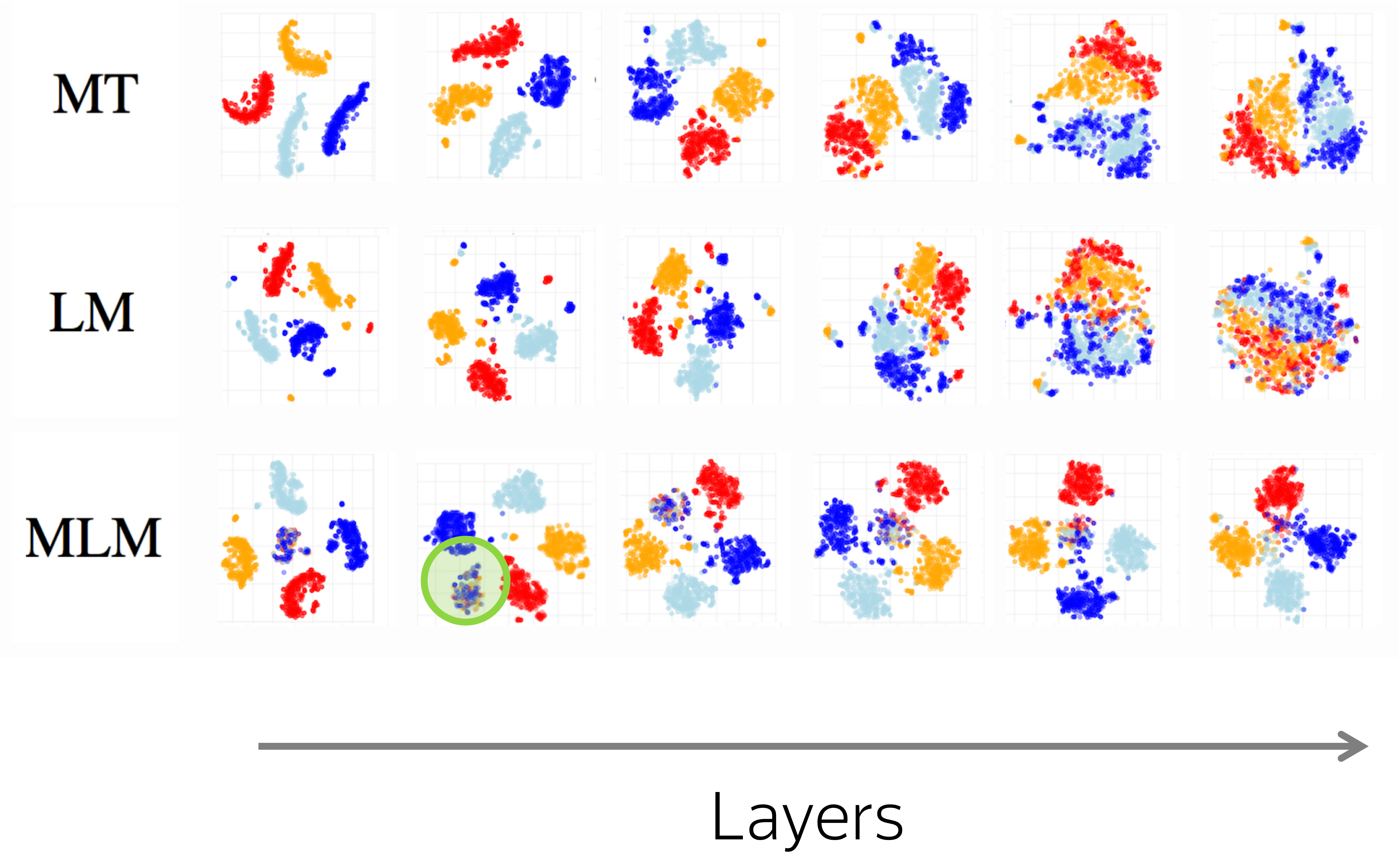
- t-SNE of different occurrences of the tokens is, are, was, were



Look how MLM disambiguates masked tokens

Preserving token identity

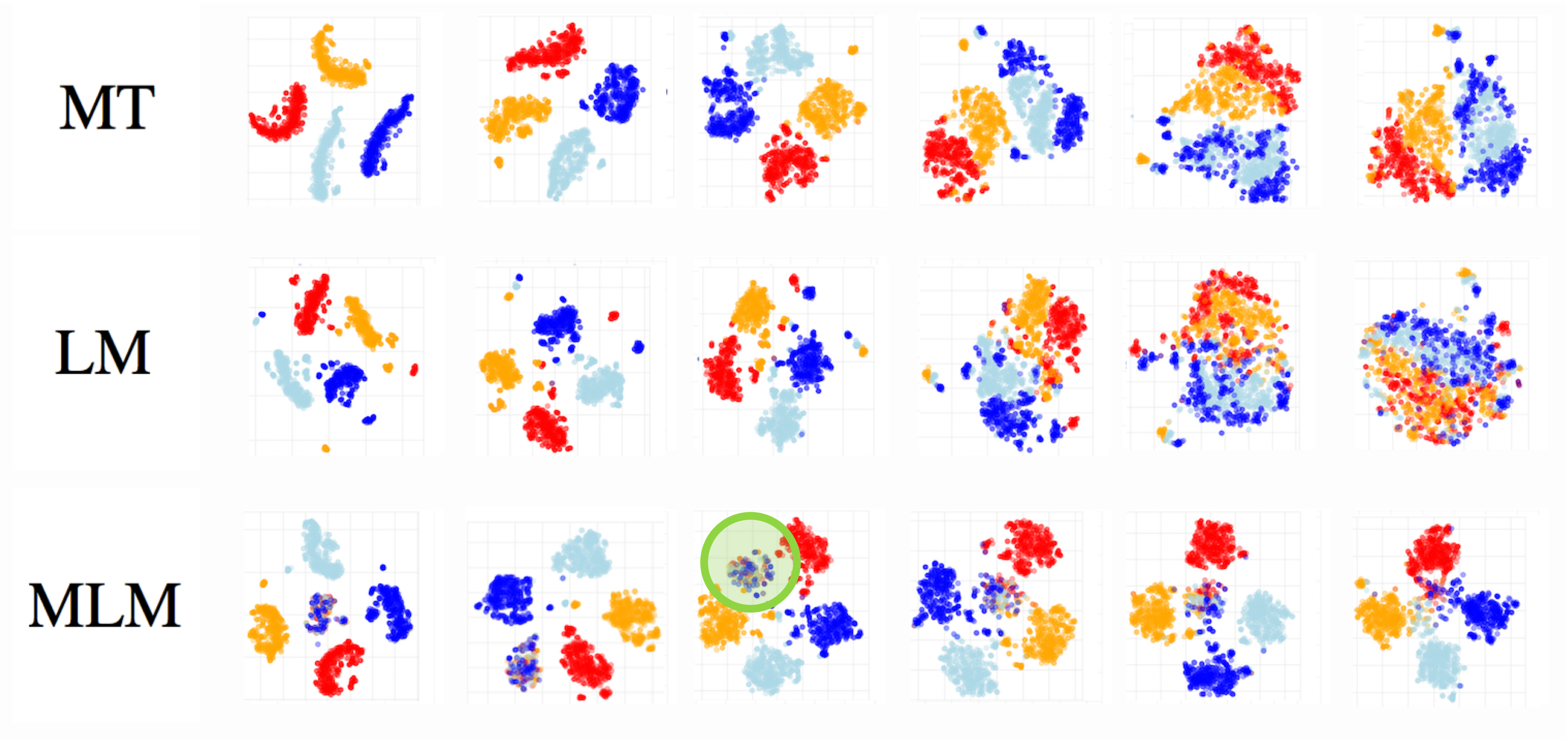
- t-SNE of different occurrences of the tokens is, are, was, were



Look how MLM disambiguates masked tokens

Preserving token identity

- t-SNE of different occurrences of the tokens is, are, was, were

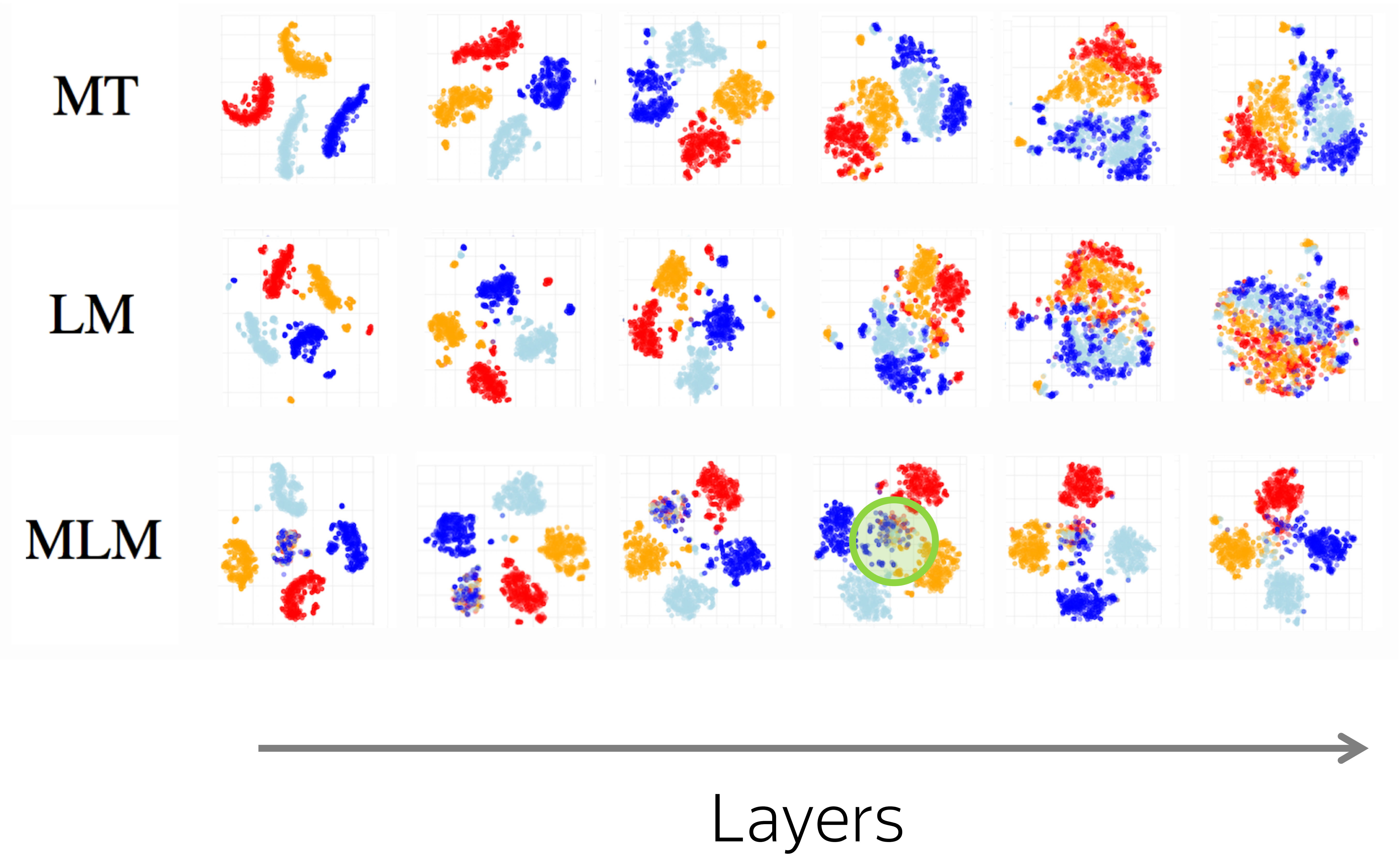


Layers

Look how MLM disambiguates masked tokens

Preserving token identity

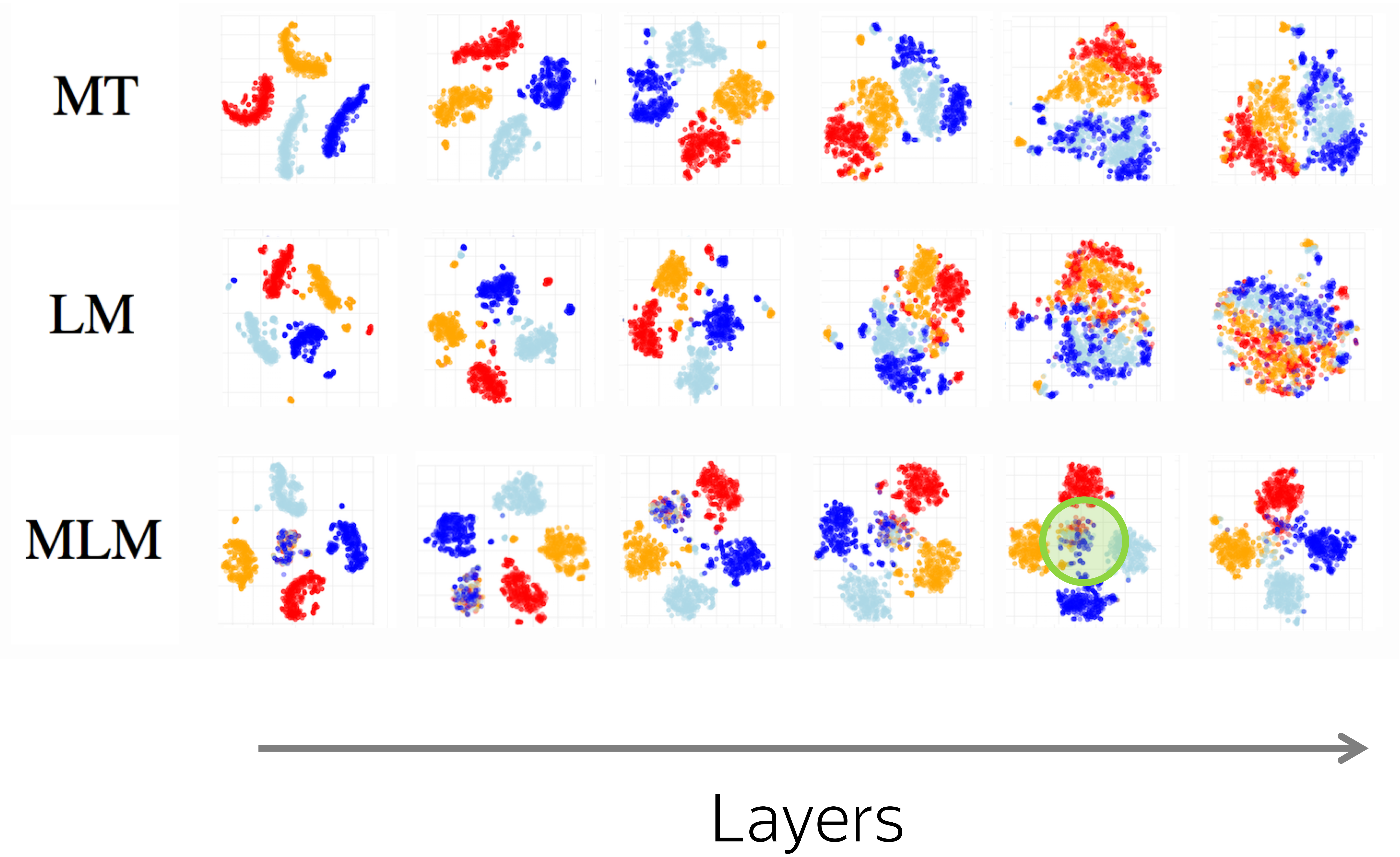
- t-SNE of different occurrences of the tokens is, are, was, were



Look how MLM disambiguates masked tokens

Preserving token identity

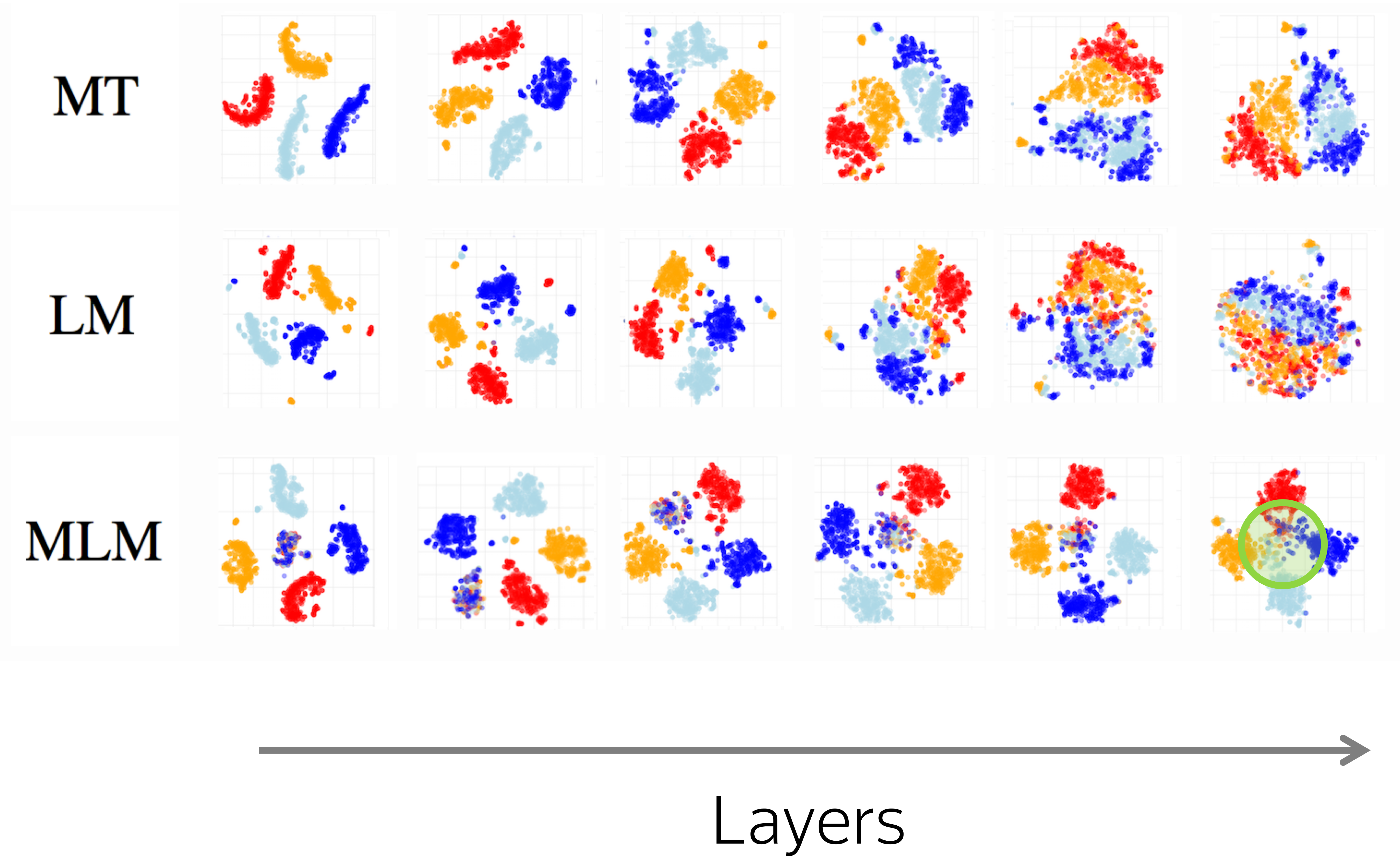
- t-SNE of different occurrences of the tokens is, are, was, were



Look how MLM disambiguates masked tokens

Preserving token identity

- t-SNE of different occurrences of the tokens is, are, was, were



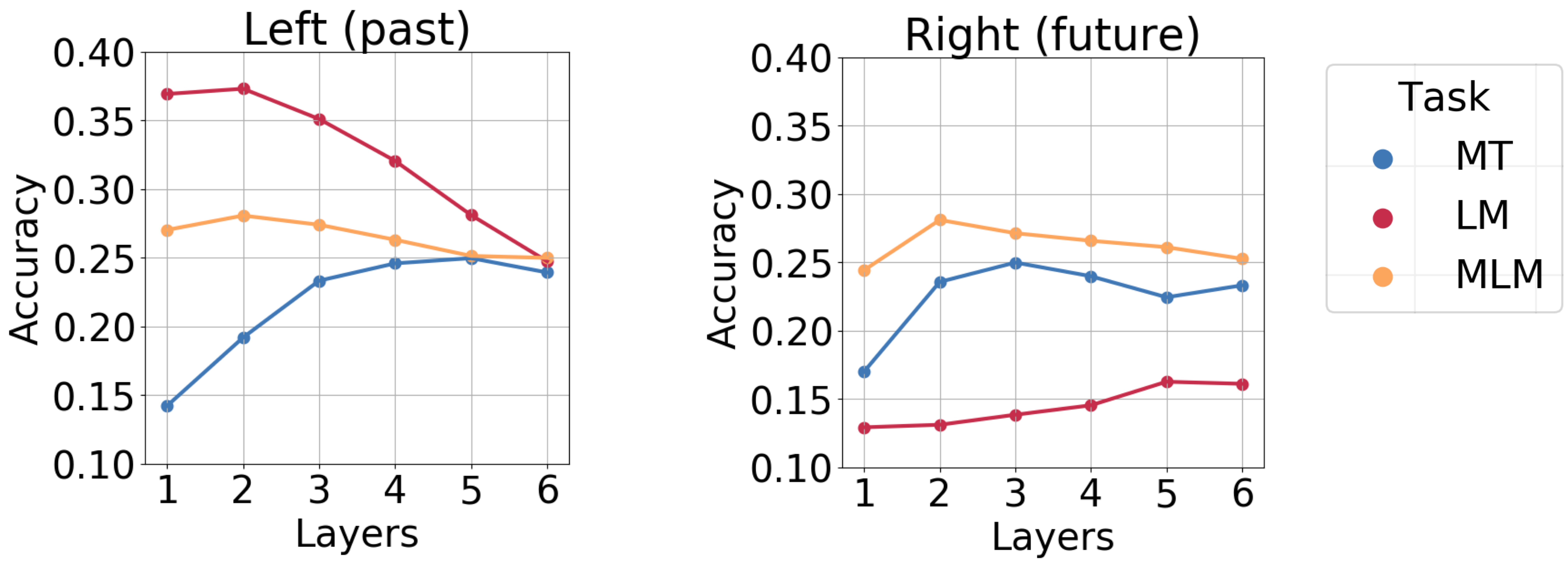
Look how MLM disambiguates masked tokens

What’s next: lexical and syntactic context

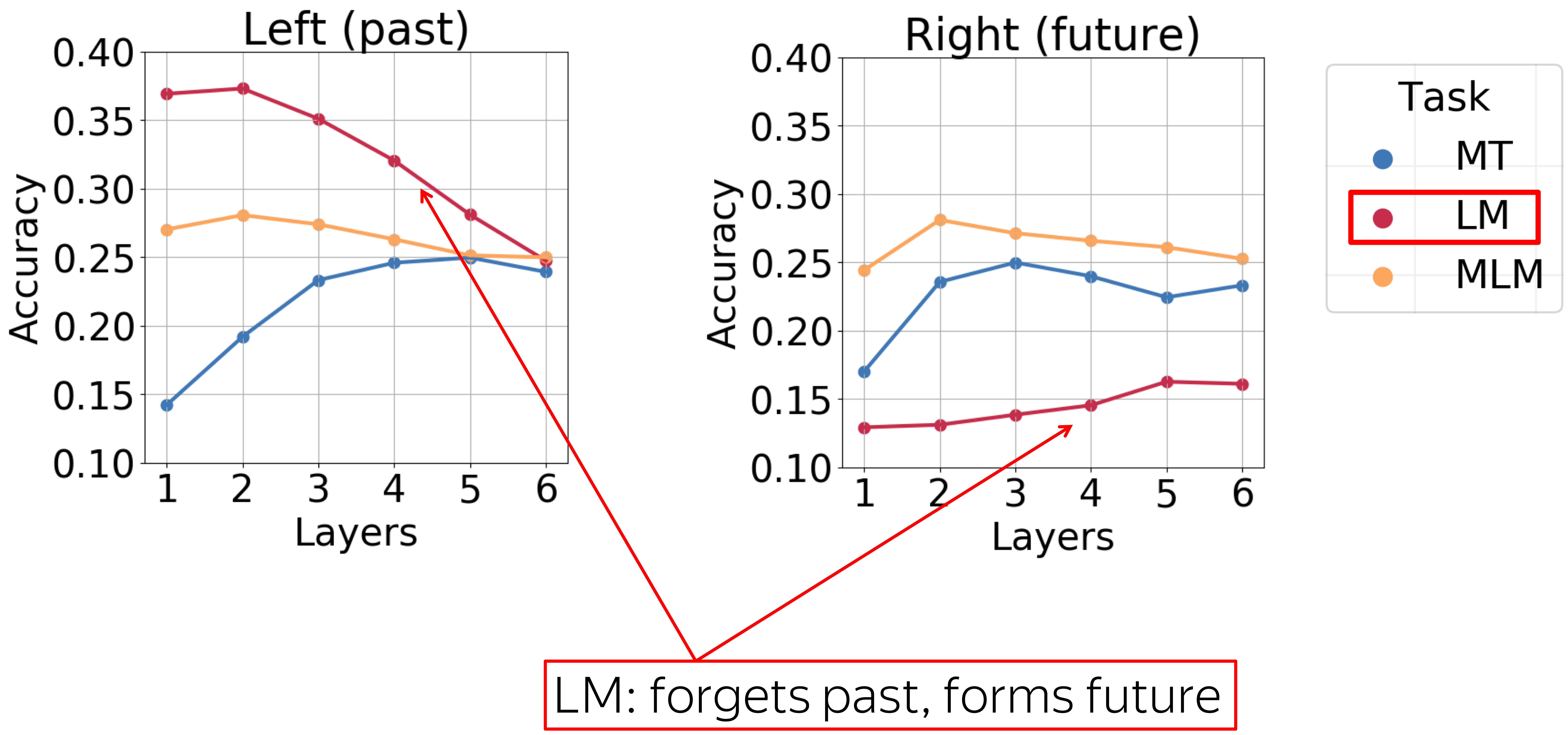
We also look at:

- Lexical context (identities of adjacent tokens)
- Syntactic context (CCG tags with their left/right parts)

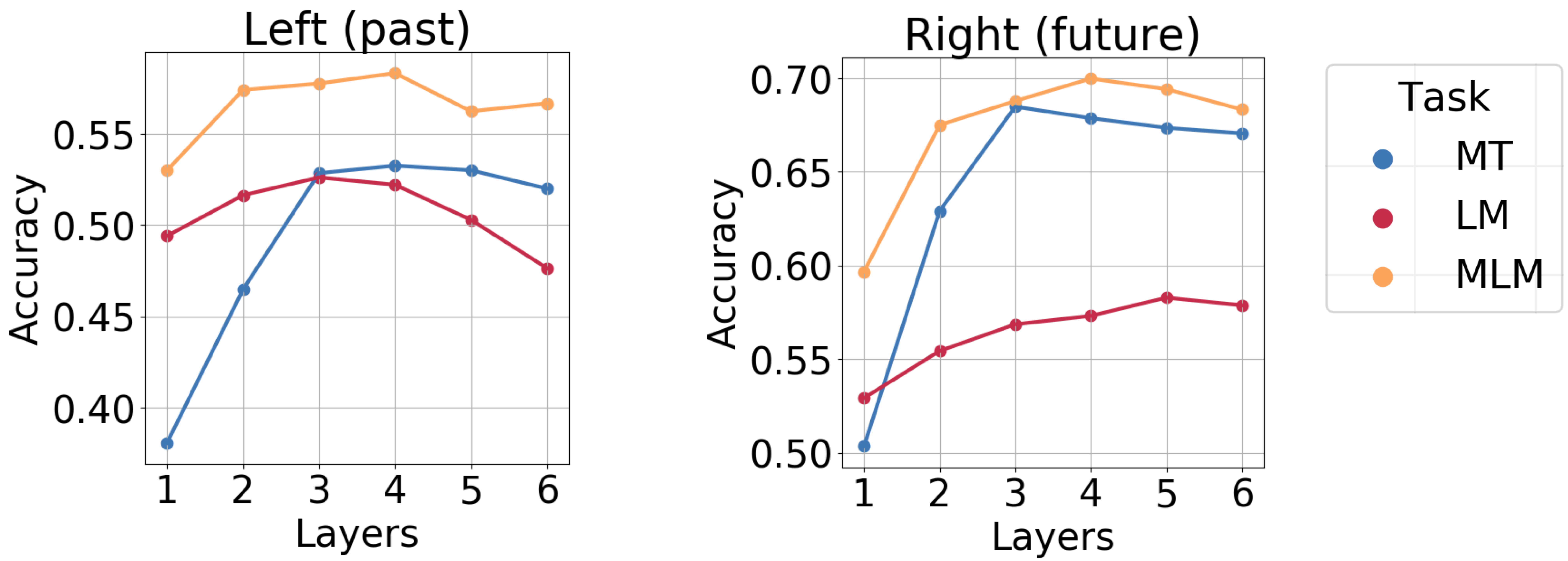
Lexical context (identities of adjacent tokens)



Lexical context (identities of adjacent tokens)

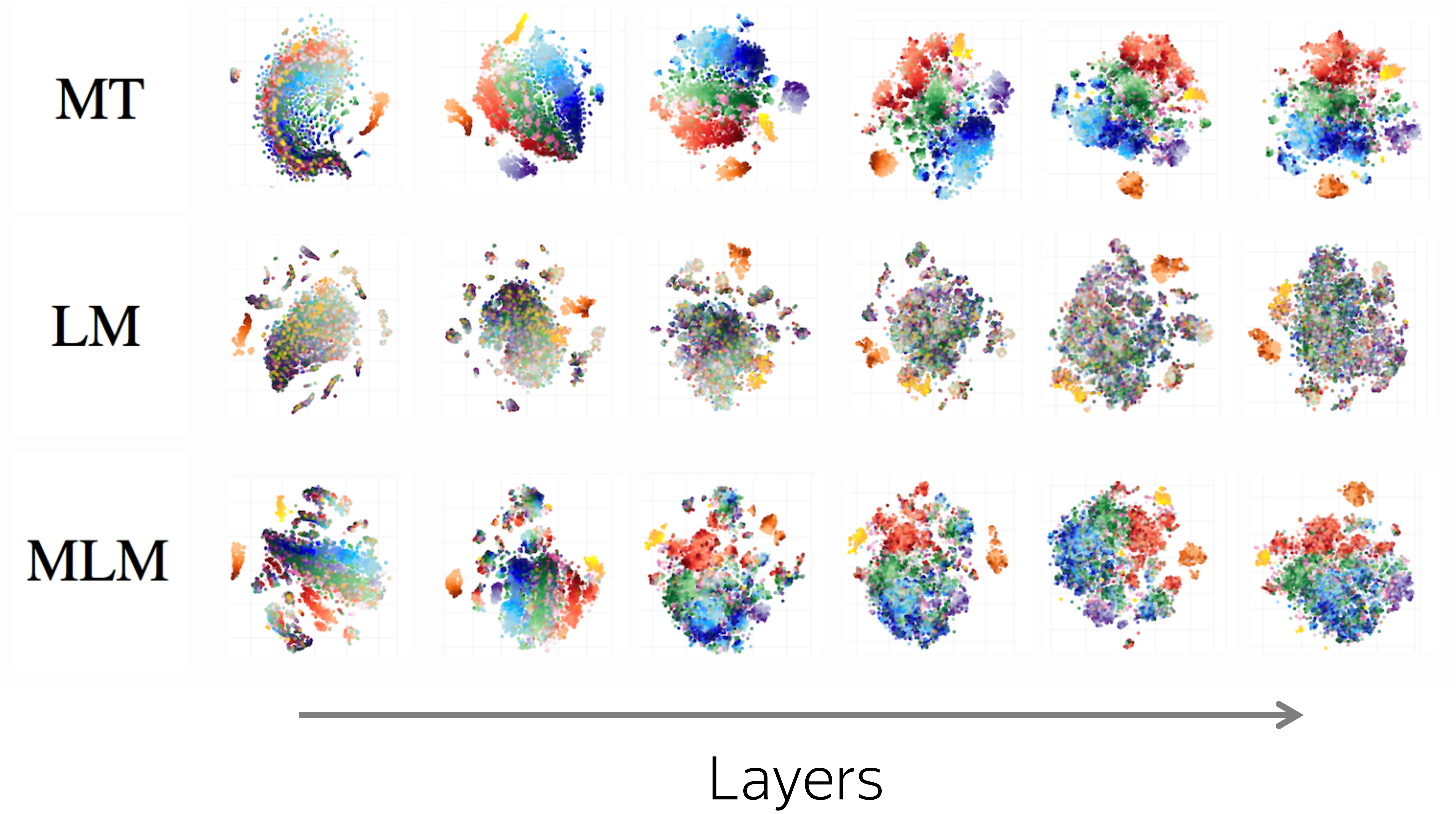


Syntactic context (CCG tags)



Syntactic context (CCG tags)

- t-SNE of different occurrences of the token “is”. CCG tag is in color.



Relation to other works

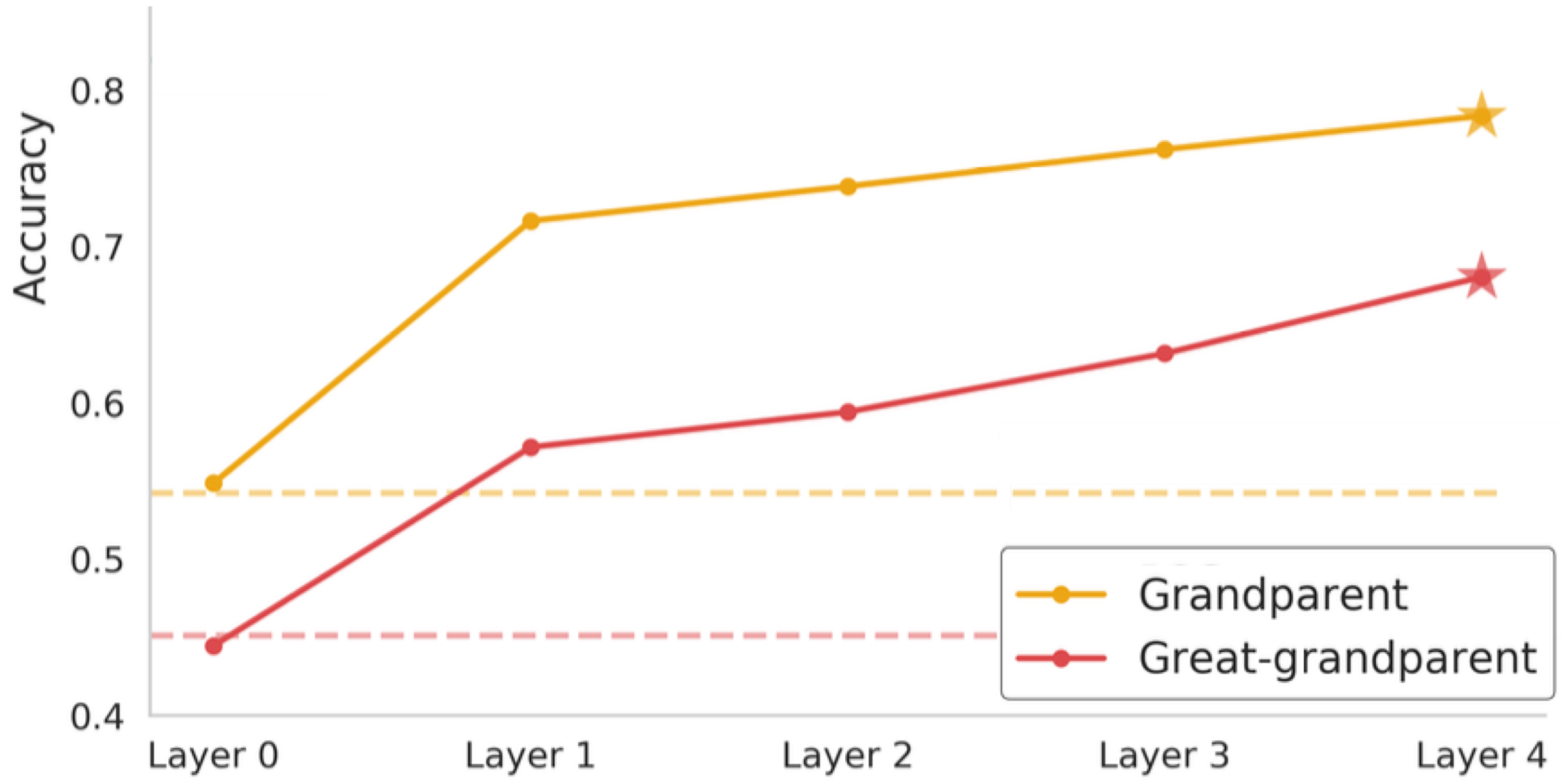
Previous work:
Untrained LSTMs are better for token prediction

- Untrained LSTMs outperform trained ones for word identity prediction task (Zhang & Bowman, 2018)

Previous work:
MT behavior is monotonic, LM is not

- For constituent labeling prediction, MT shows monotonic behavior, while LM non-monotonic (Blevins et al, 2018)

MT



LM

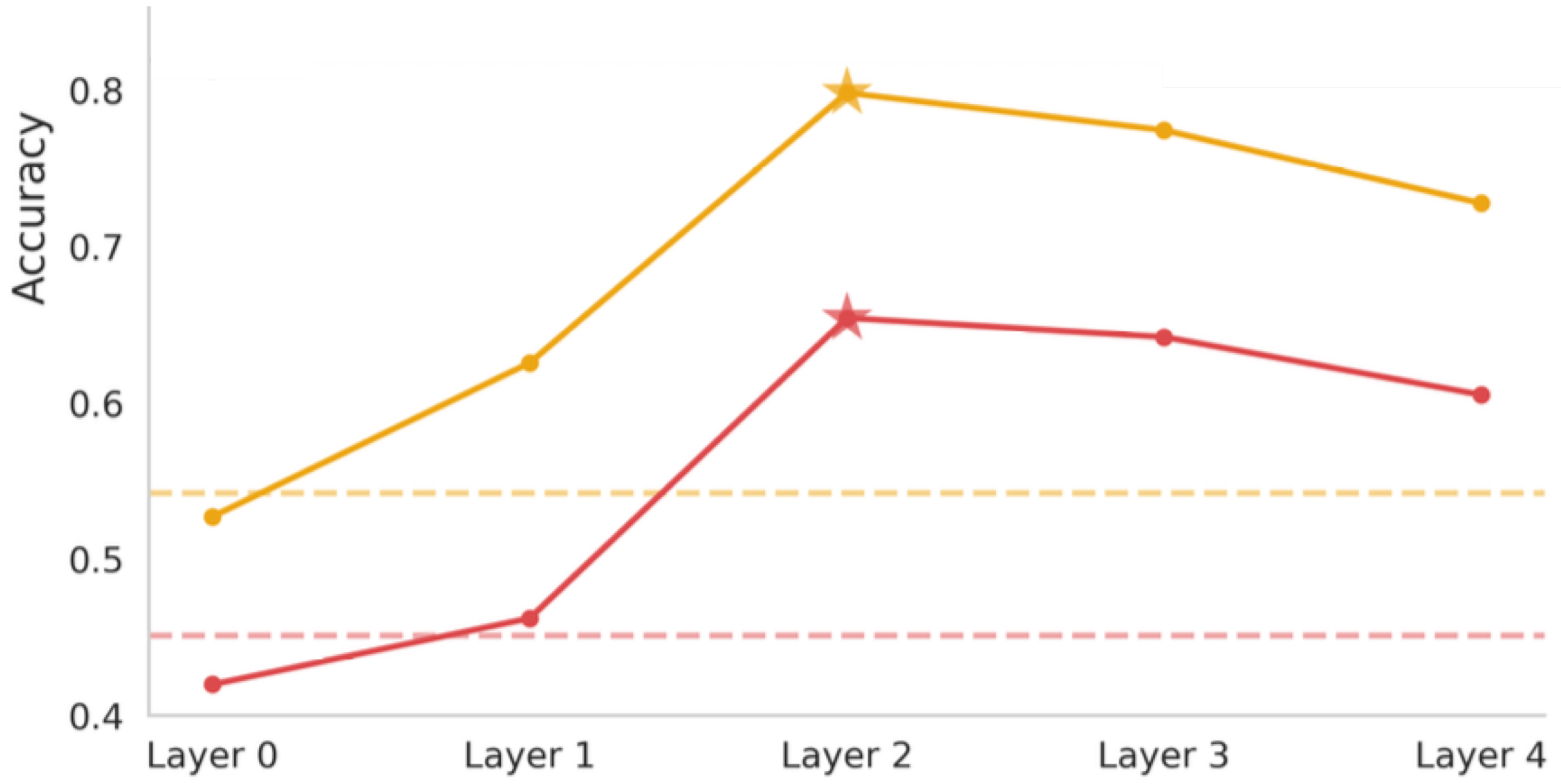


Illustration is from the original paper by Blevins et al, 2018

Previous work: BERT behavior is not monotonic

- For different tasks the contribution of a layer to a task increases up to a certain layer, but then decreases at the top layers (Tenney et al, 2019)

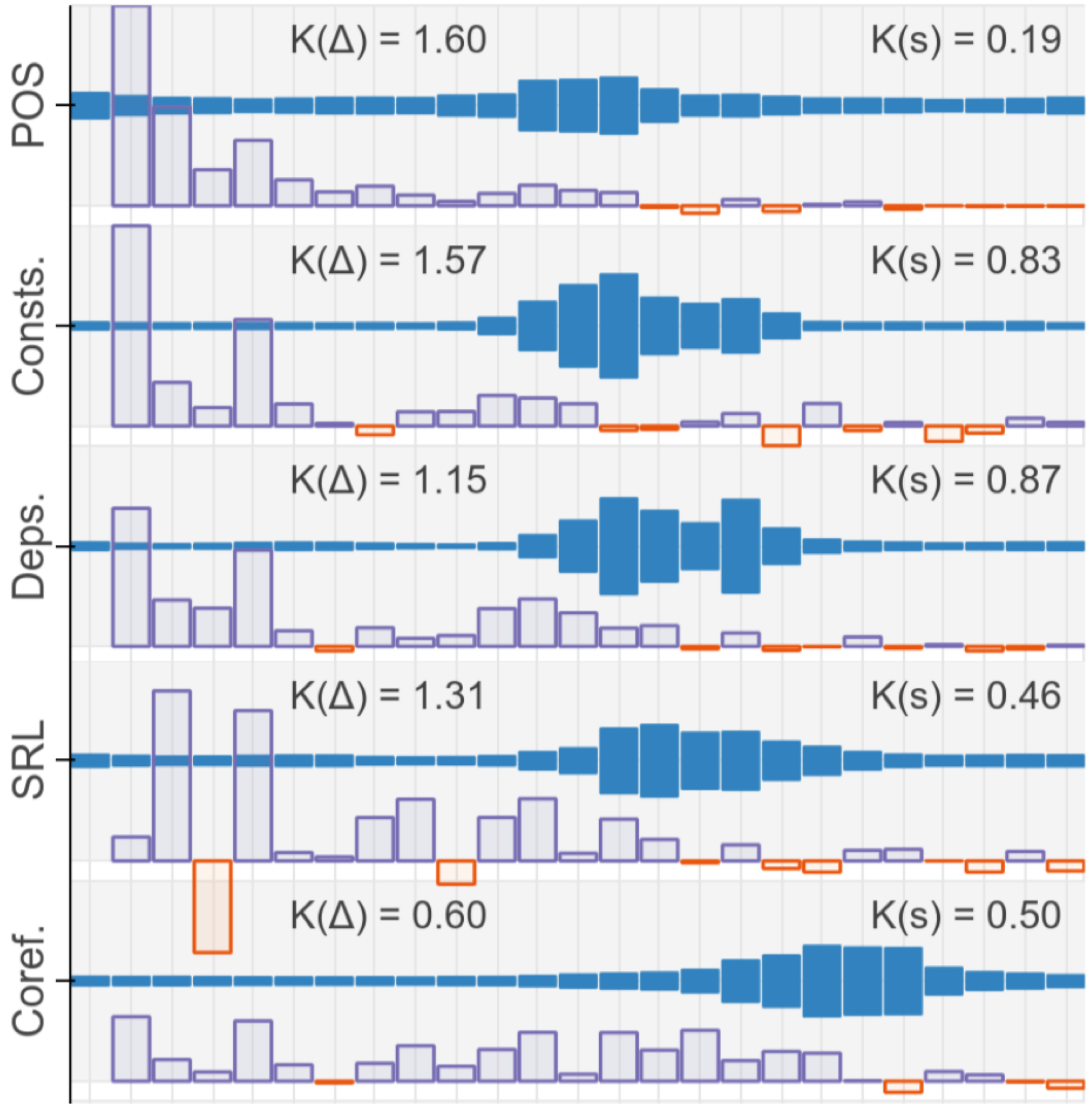
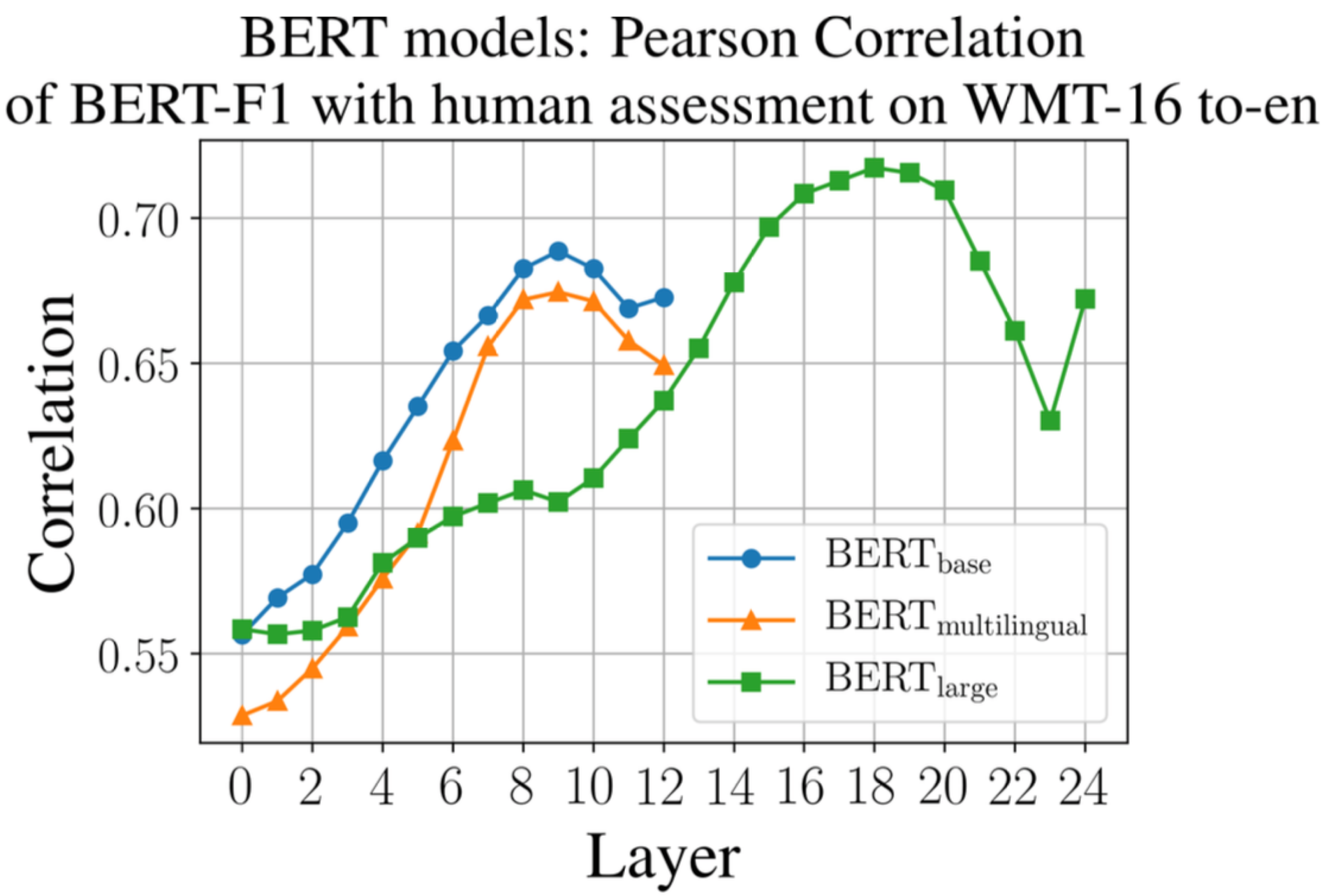


Illustration is from the original paper by Tenney et al, 2019

Recent works
BERTScore: Evaluating Text Generation with BERT

(Tianyi Zhang*, Varsha Kishore*, Felix Wu*, Kilian Q. Weinberger, Yoav Artzi, ICLR 2020)

- BERT representations are used to build a metric



The two stages: ’context encoding’ and ‘token reconstruction’

Illustration is from the original paper

Conclusions

Our key findings are:

- for LM, evolution is a transition from known past to the unknown future;

- MLMs initially acquire information about context, then recreate token; this happens in two stages;

- for MT, representations get refined with context, but most of the information is preserved.

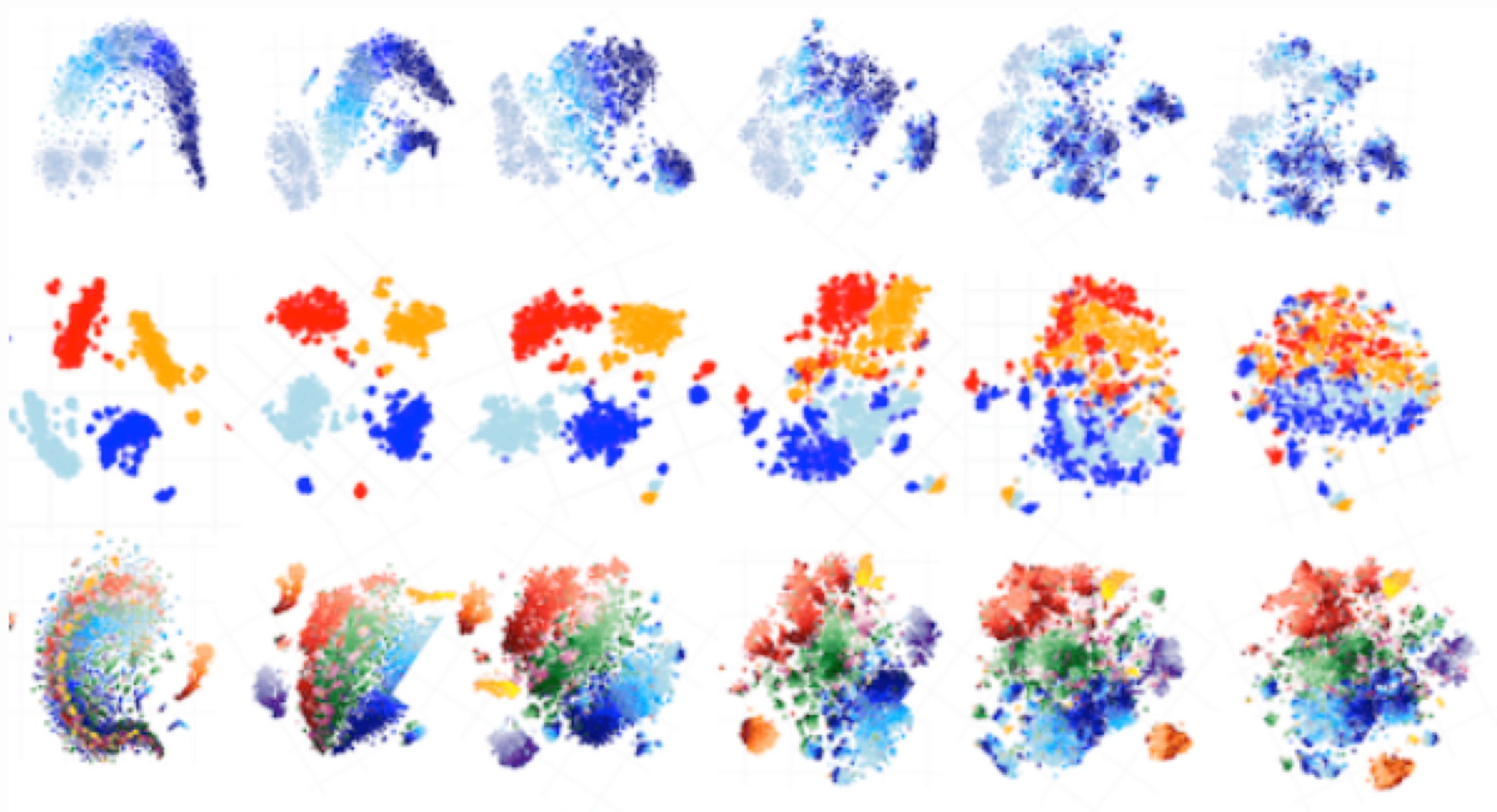
Our key contributions:

- we propose to view the evolution of a token representation from the compression/prediction trade-off perspective;
- we conduct a series of experiments supporting this view;
- we relate to some findings from previous work, putting them in the proposed perspective.

Official blog post

Evolution of Representations in the Transformer

This is a post for the EMNLP 2019 paper The Bottom-up Evolution of Representations in the Transformer: A Study with Machine Translation and Language Modeling Objectives.



We look at the evolution of representations of individual tokens in Transformers trained with different training objectives (MT, LM, MLM - BERT-style) from the Information Bottleneck perspective and show, that:

- LMs gradually forget past when forming predictions about future;
- for MLMs, the evolution proceeds in two stages of **context encoding** and **token reconstruction**;
- MT representations get refined with context, but less processing is happening.

read more | read paper

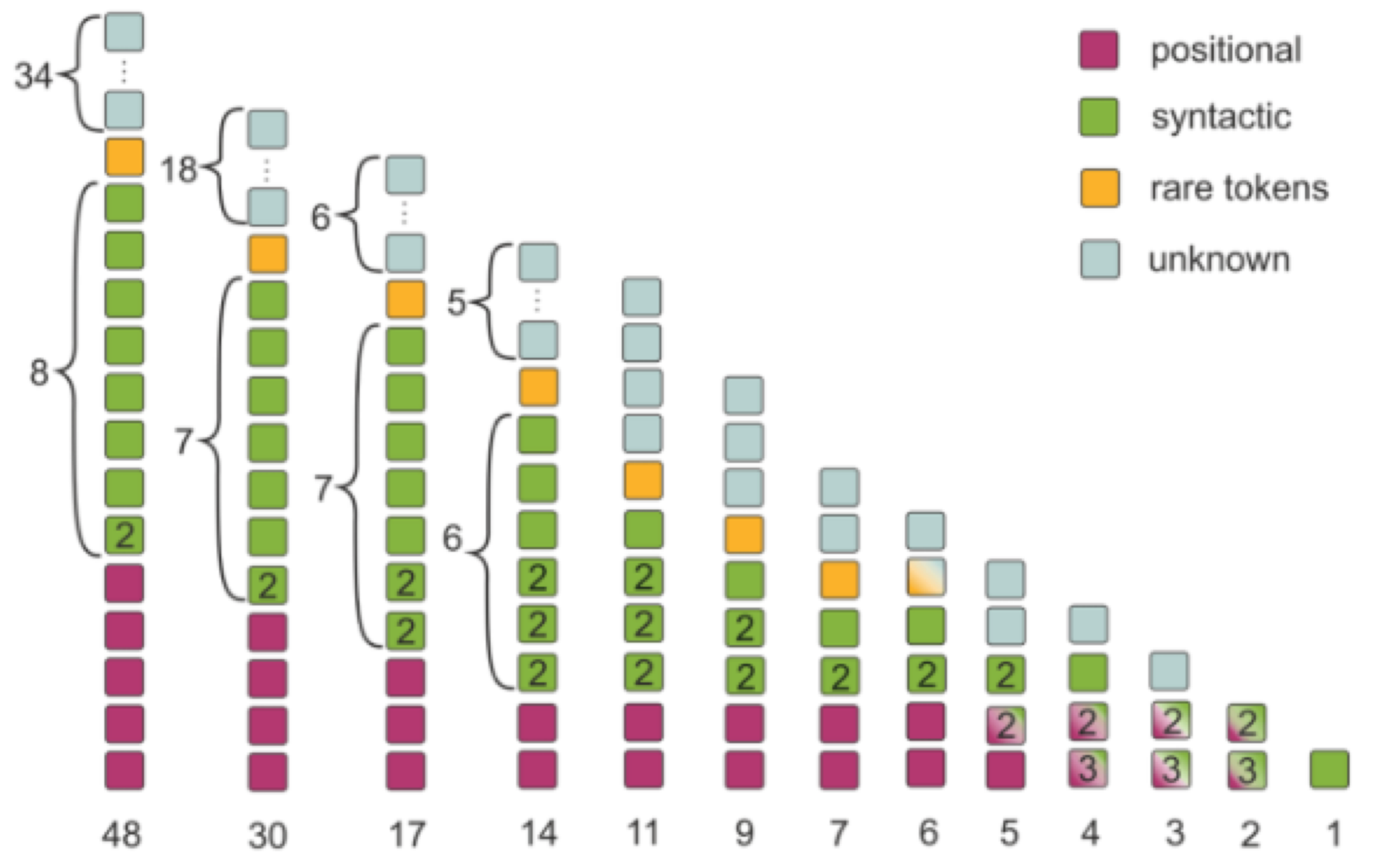
September 2019

https://lena-voita.github.io

More Analysis: The Story of Heads

The Story of Heads

This is a post for the ACL 2019 paper Analyzing Multi-Head Self-Attention: Specialized Heads Do the Heavy Lifting, the Rest Can Be Pruned.



From this post, you will learn:

- how we evaluate the importance of attention heads in Transformer
- which functions the most important encoder heads perform
- how we prune the vast majority of attention heads in Transformer without seriously affecting quality
- which types of model attention are most sensitive to the number of attention heads and on which layers

→≡ read more | read paper | </> view code

June 2019

https://lena-voita.github.io

Thank you!

Lena Voita

Research Scientist, Yandex Research

PhD student, Uni Amsterdam & Uni Edinburgh

lena-voita@yandex-team.ru

https://lena-voita.github.io

@lena_voita

lena-voita