

# Learning to edit pre-trained models

*CS330: Frontiers and Open Challenges*

# Editing Neural Nets: Why?

Neural networks contain lots of “knowledge”

# Editing Neural Nets: Why?

Neural networks contain lots of “knowledge”, but...

...models can be wrong, or become obsolete over time!

# Editing Neural Nets: Why?

Neural networks contain lots of “knowledge”, but...

...models can be wrong, or become obsolete over time!

Large ~SOTA question-answering model:

**Input: Who is the prime minister of the UK?**

*Output: Theresa May* ← Not anymore!

# Editing Neural Nets: Why?

Neural networks contain lots of “knowledge”, but...

...models can be wrong, or become obsolete over time!

Large ~SOTA question-answering model:

**Input: Who is the prime minister of the UK?**

*Output: Theresa May* ← Not anymore!

How can we **locally** tweak our model’s behavior?

# Editing Neural Nets: Why?

Neural networks contain lots of “knowledge”, but...

...models can be wrong, or become obsolete over time!

Large ~SOTA question-answering model:

**Input: Who is the prime minister of the UK?**

*Output: Theresa May* ← Not anymore!

How can we **locally** tweak our model’s behavior?

Other examples:

- Image classifier errs on one particular background (snow)
- Policy screws up, but only in one particular situation
- Translation system mistranslates a particular phrase
- ...

The model is *mostly* right; how do we change its behavior just for this example (and related examples)?

# Editing Neural Nets: How?

This is one-shot learning... sort of?

# Editing Neural Nets: How?

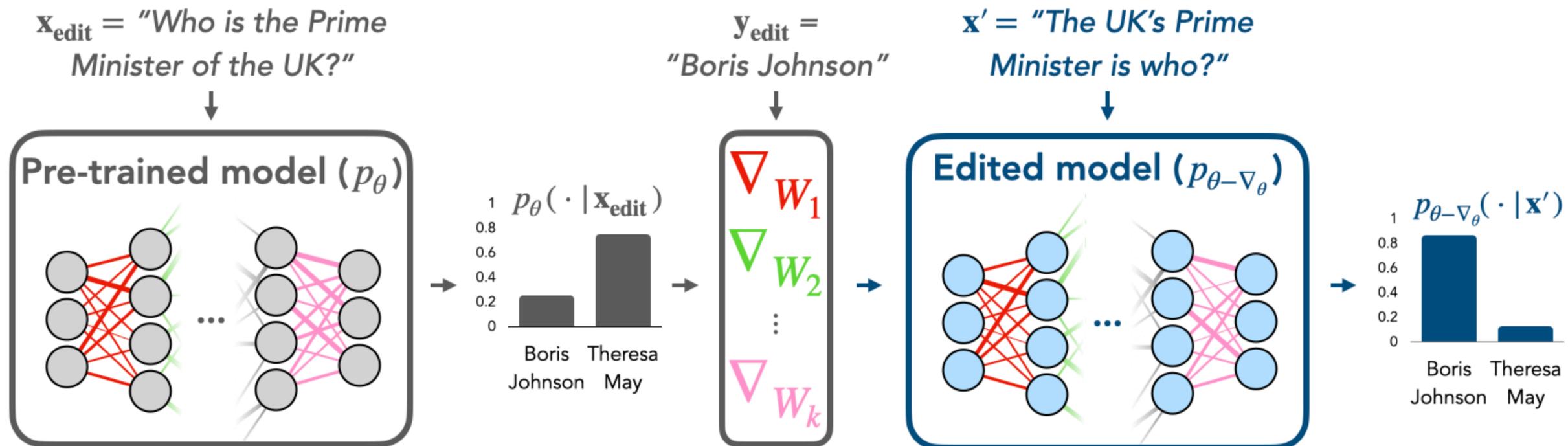
This is one-shot learning... sort of?

**Assume:** (meta-)dataset of questions  $(\mathbf{x}_{\text{edit}}, y_{\text{edit}}, \mathbf{x}')_i$  [each question is a different 'task']

# Editing Neural Nets: How?

This is one-shot learning... sort of?

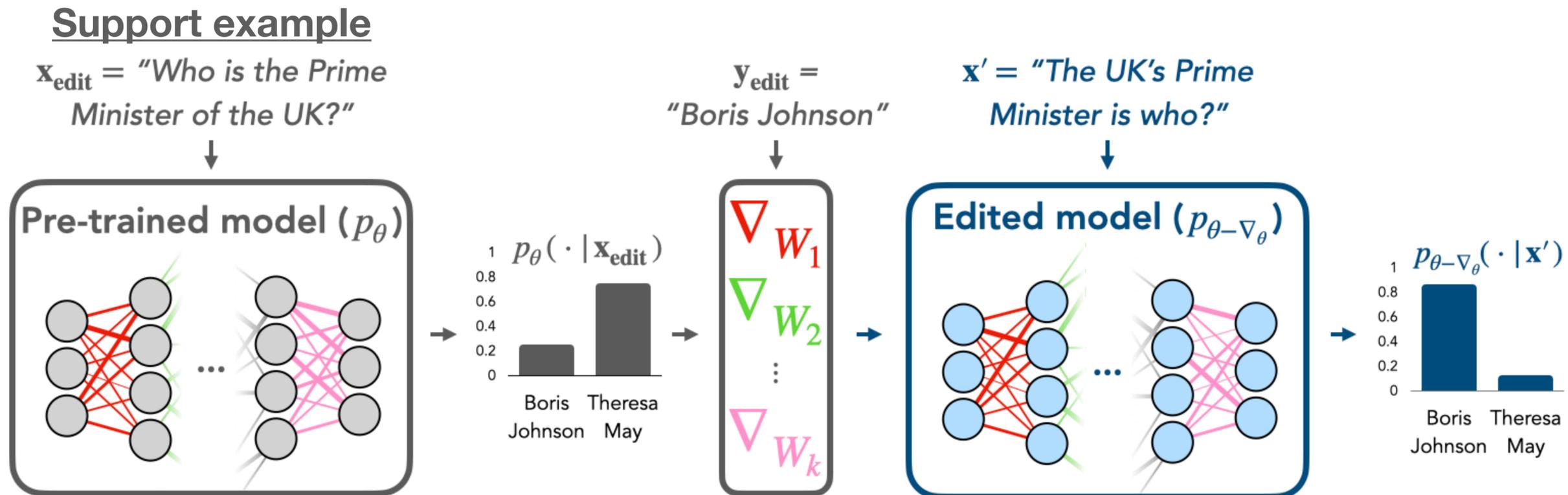
**Assume:** (meta-)dataset of questions  $(\mathbf{x}_{\text{edit}}, y_{\text{edit}}, \mathbf{x}')_i$  [each question is a different 'task']



# Editing Neural Nets: How?

This is one-shot learning... sort of?

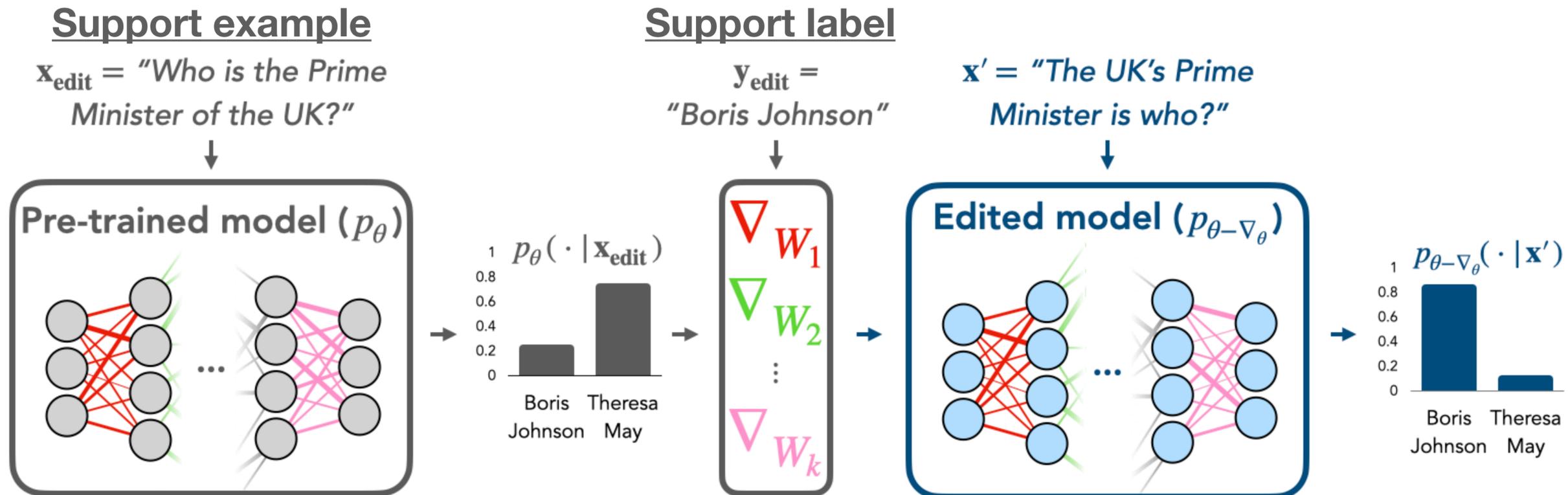
**Assume:** (meta-)dataset of questions  $(\mathbf{x}_{\text{edit}}, y_{\text{edit}}, \mathbf{x}')_i$  [each question is a different 'task']



# Editing Neural Nets: How?

This is one-shot learning... sort of?

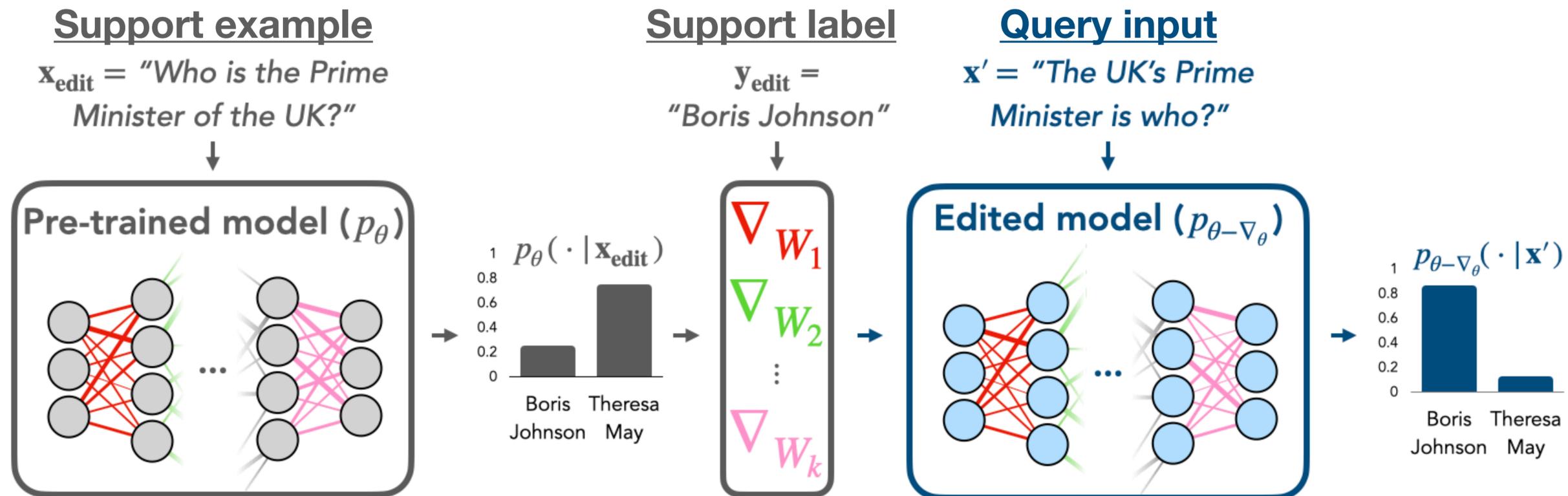
**Assume:** (meta-)dataset of questions  $(\mathbf{x}_{\text{edit}}, y_{\text{edit}}, \mathbf{x}')_i$  [each question is a different 'task']



# Editing Neural Nets: How?

This is one-shot learning... sort of?

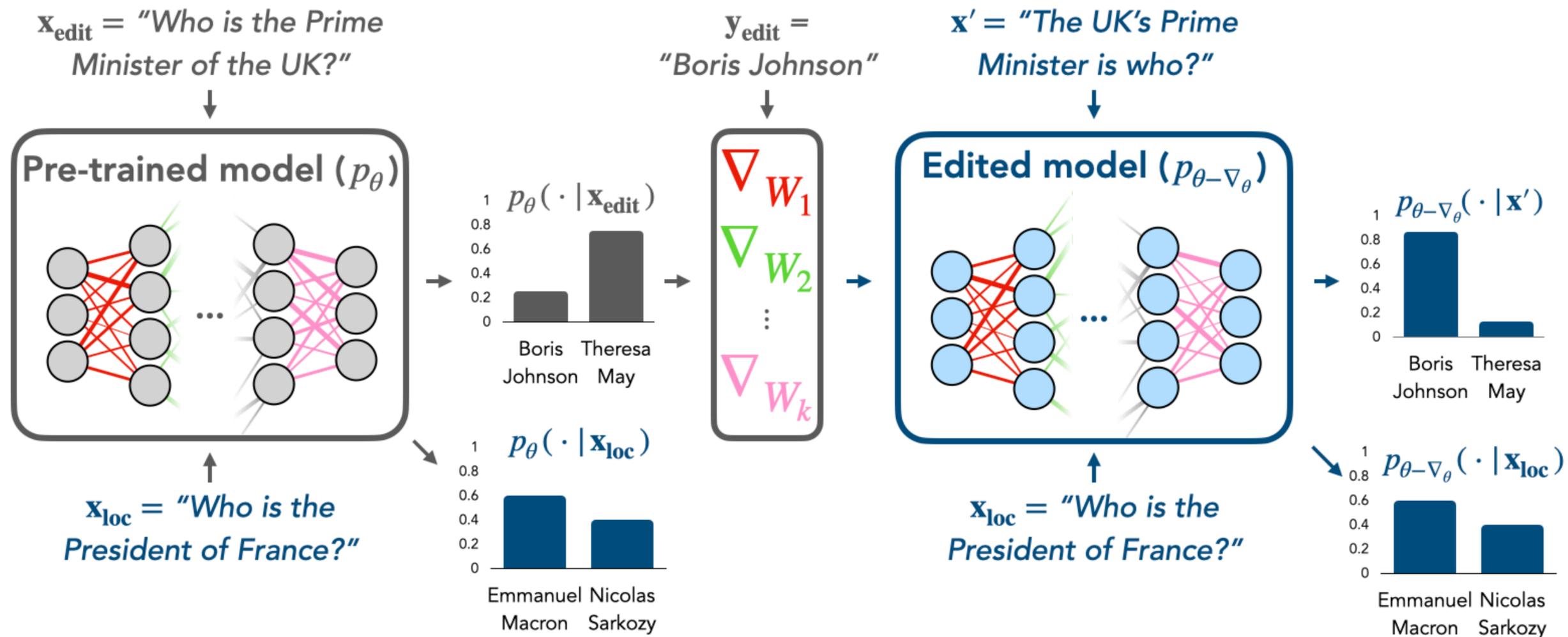
**Assume:** (meta-)dataset of questions  $(\mathbf{x}_{\text{edit}}, y_{\text{edit}}, \mathbf{x}')_i$  [each question is a different 'task']



# Editing Neural Nets: How?

This is one-shot learning... sort of?

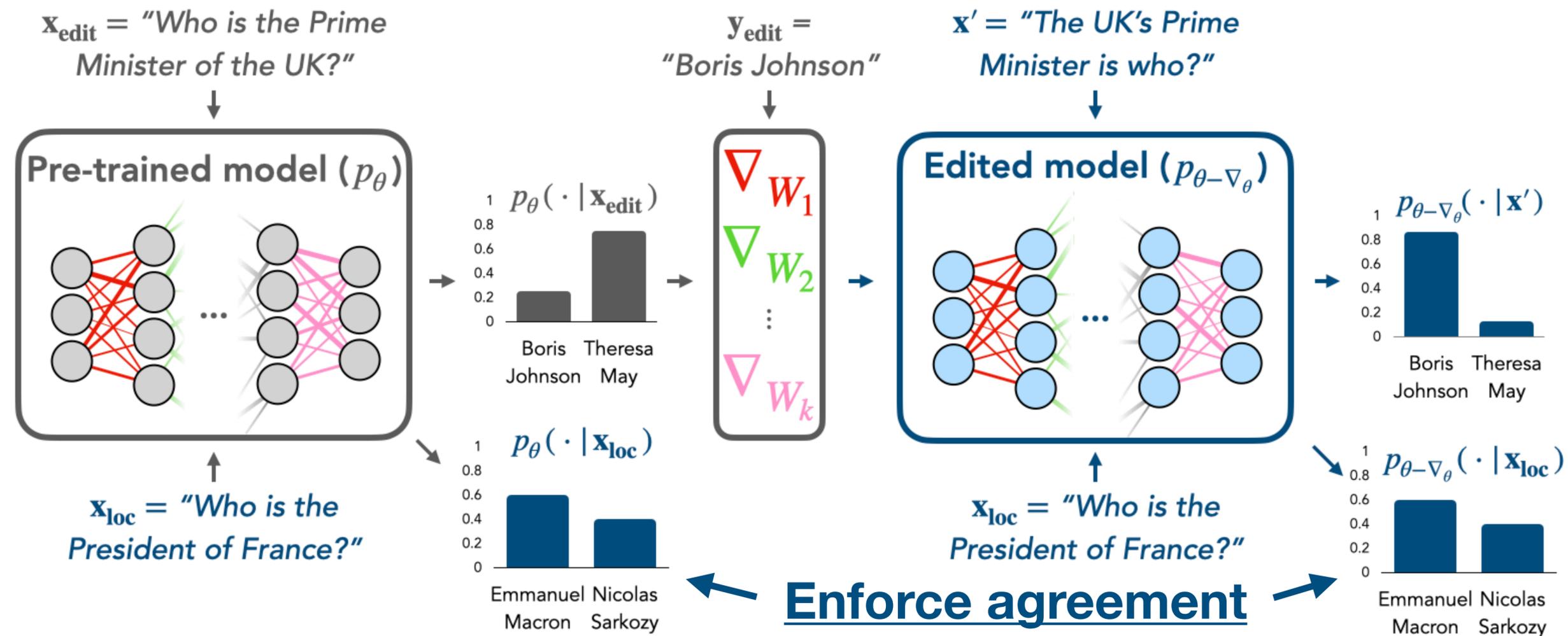
**Assume:** (meta-)dataset of questions  $(\mathbf{x}_{\text{edit}}, y_{\text{edit}}, \mathbf{x}')_i$  [each question is a different 'task']



# Editing Neural Nets: How?

This is one-shot learning... sort of?

**Assume:** (meta-)dataset of questions  $(\mathbf{x}_{\text{edit}}, \mathbf{y}_{\text{edit}}, \mathbf{x}')_i$  [each question is a different 'task']



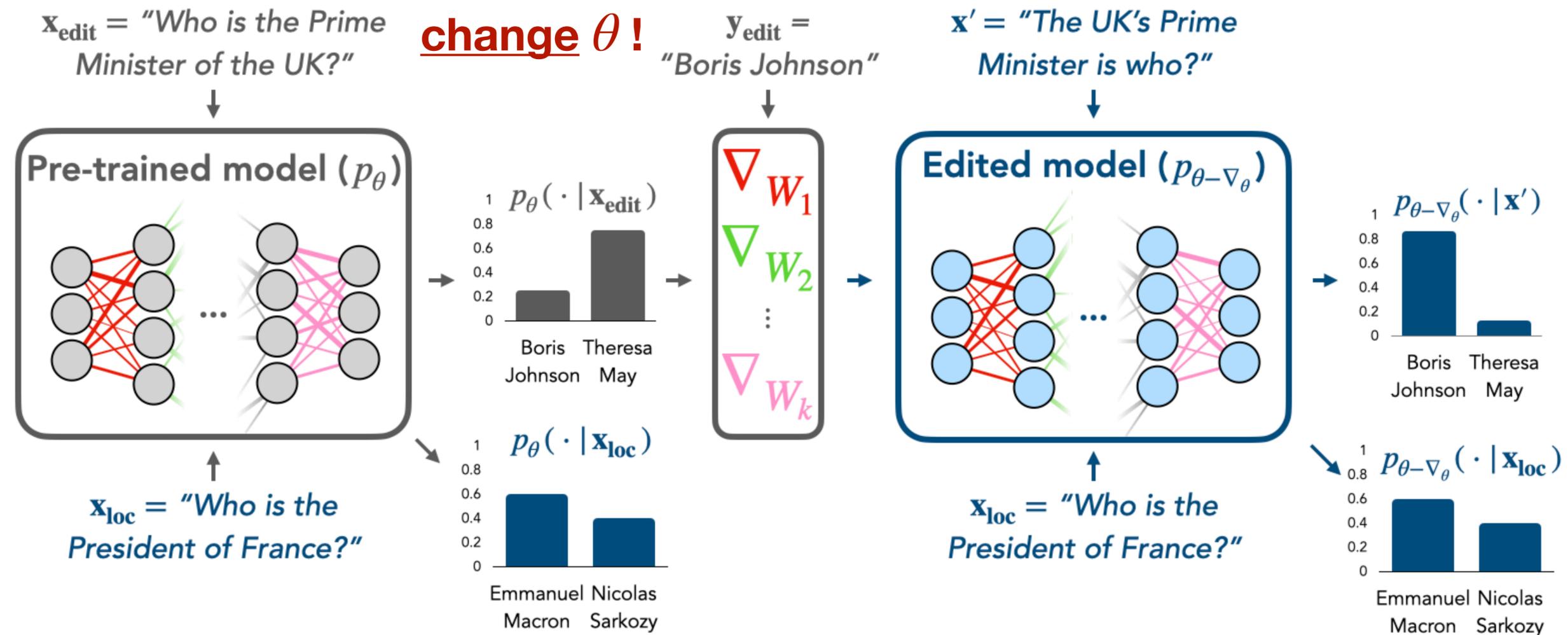
# Editing Neural Nets: How?

This is one-shot learning... sort of?

**Assume:** (meta-)dataset of questions  $(\mathbf{x}_{\text{edit}}, y_{\text{edit}}, \mathbf{x}')_i$  [each question is a different 'task']

**Don't want to**

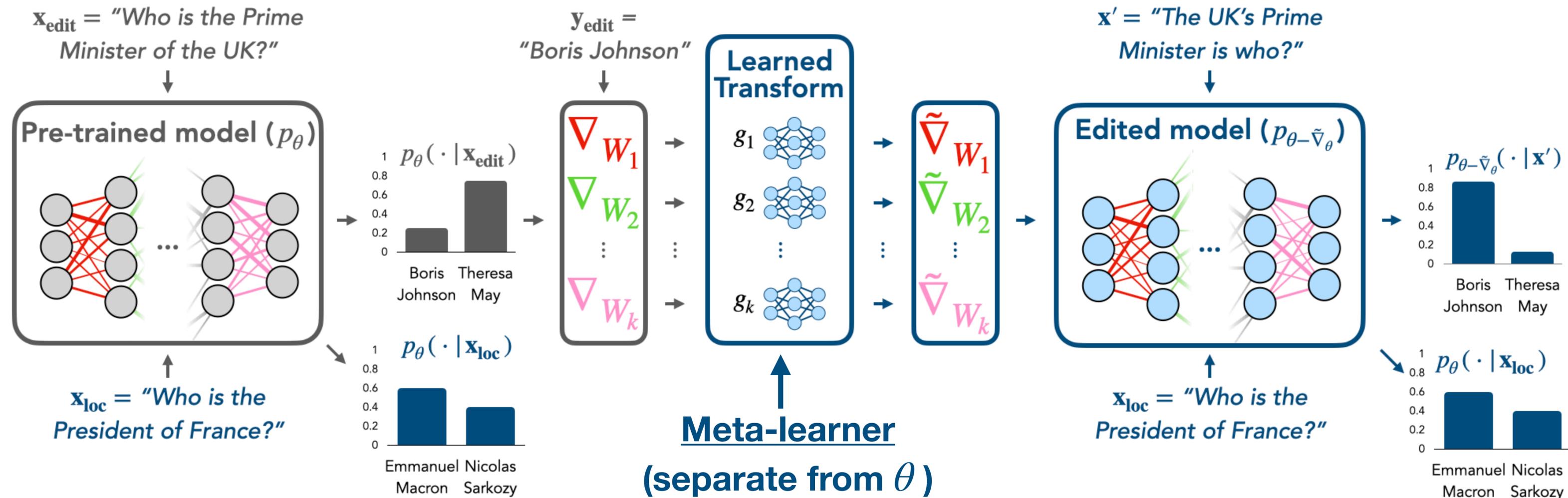
**change  $\theta$  !**



# Editing Neural Nets: How?

This is one-shot learning... sort of?

**Assume:** (meta-)dataset of questions  $(\mathbf{x}_{\text{edit}}, \mathbf{y}_{\text{edit}}, \mathbf{x}')_i$  [each question is a different 'task']



# Recap

Editing models lets us locally adjust their behavior after training

Editing is *sort of* like one-shot learning, except:

- During adaptation, only want to change predictions *locally*
- We're given a pre-trained model & *shouldn't change the initialization*

If interested, check out *Fast Model Editing at Scale (on arXiv)* or reach out:

**eric.mitchell@cs.stanford.edu**