Calibrating Factual Knowledge in Pretrained Language Models

Yufang Liu

East China Normal University

Editing Neural Nets: Why?

- Neural networks contain many beliefs, but…

Input: Who is the prime minister of the UK?

T5: *Theresa May*

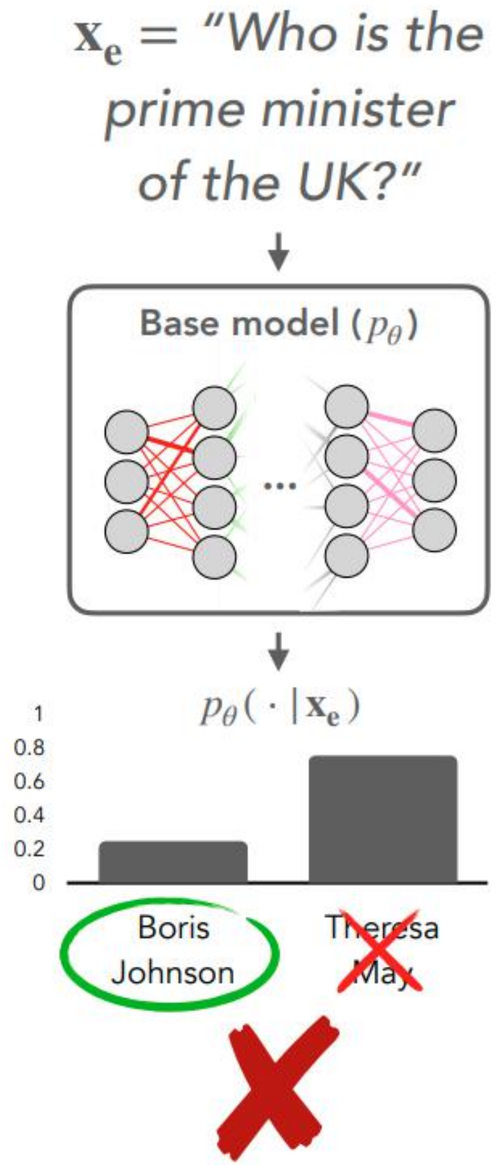
BART: *Theresa May*

GPT-3: *Theresa May*

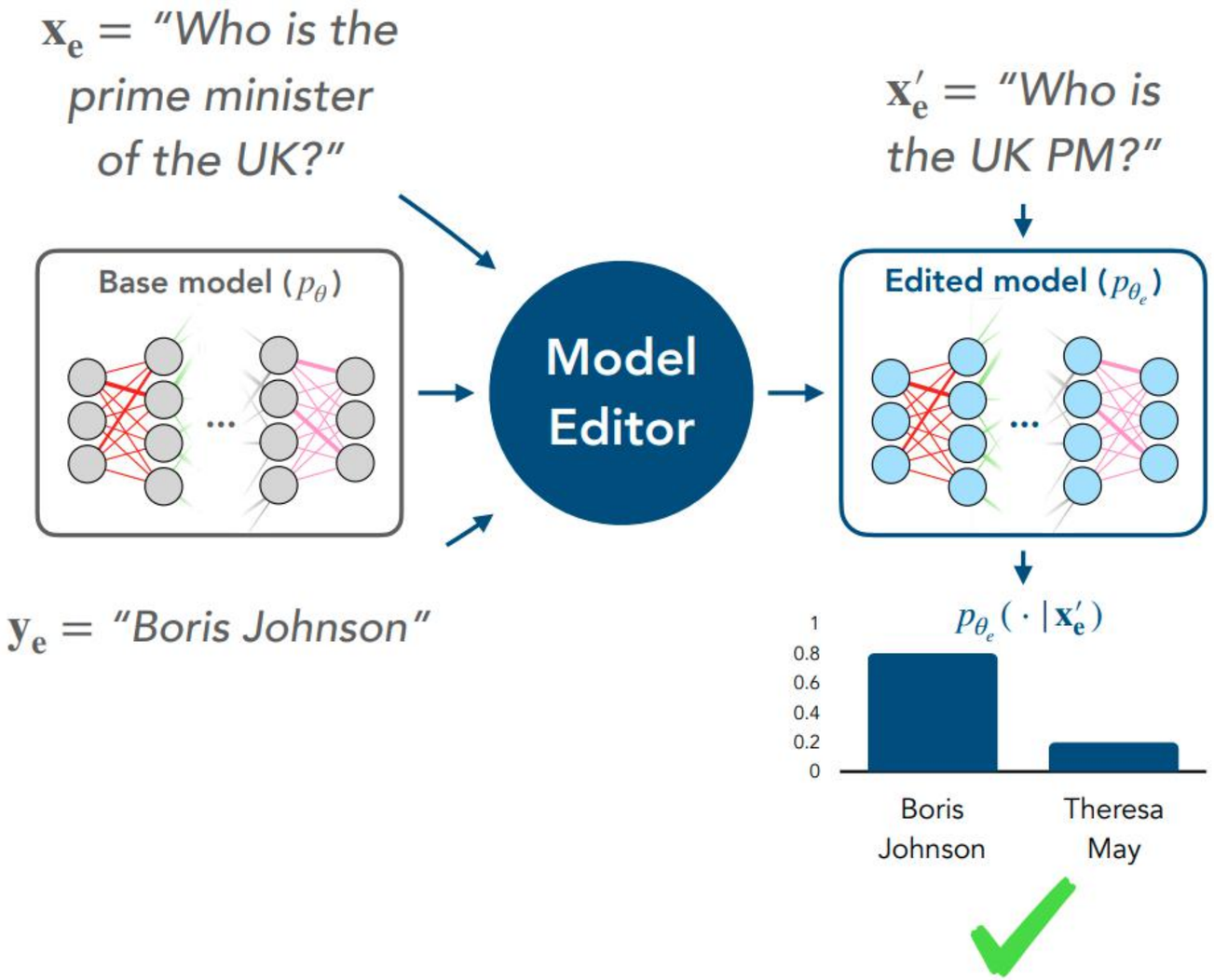
} Not anymore!

…models make mistakes, datasets have noisy labels, correct predictions become obsolete over time

Editing Neural Nets: Why?



Editing Neural Nets: Why?



semantically equivalent questions

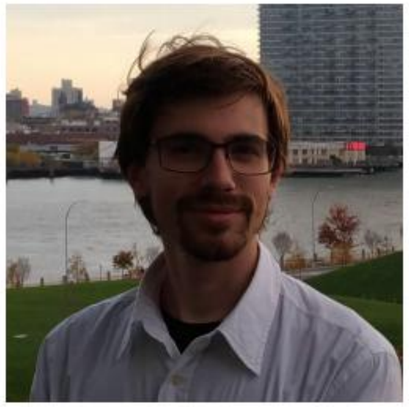
↕

unrelated questions

UNIVERSITY OF AMSTERDAM
THE UNIVERSITY of EDINBURGH

Editing Factual Knowledge in Language Models*

Nicola De Cao



Wilker Aziz

Ivan Titov

EMNLP 2021

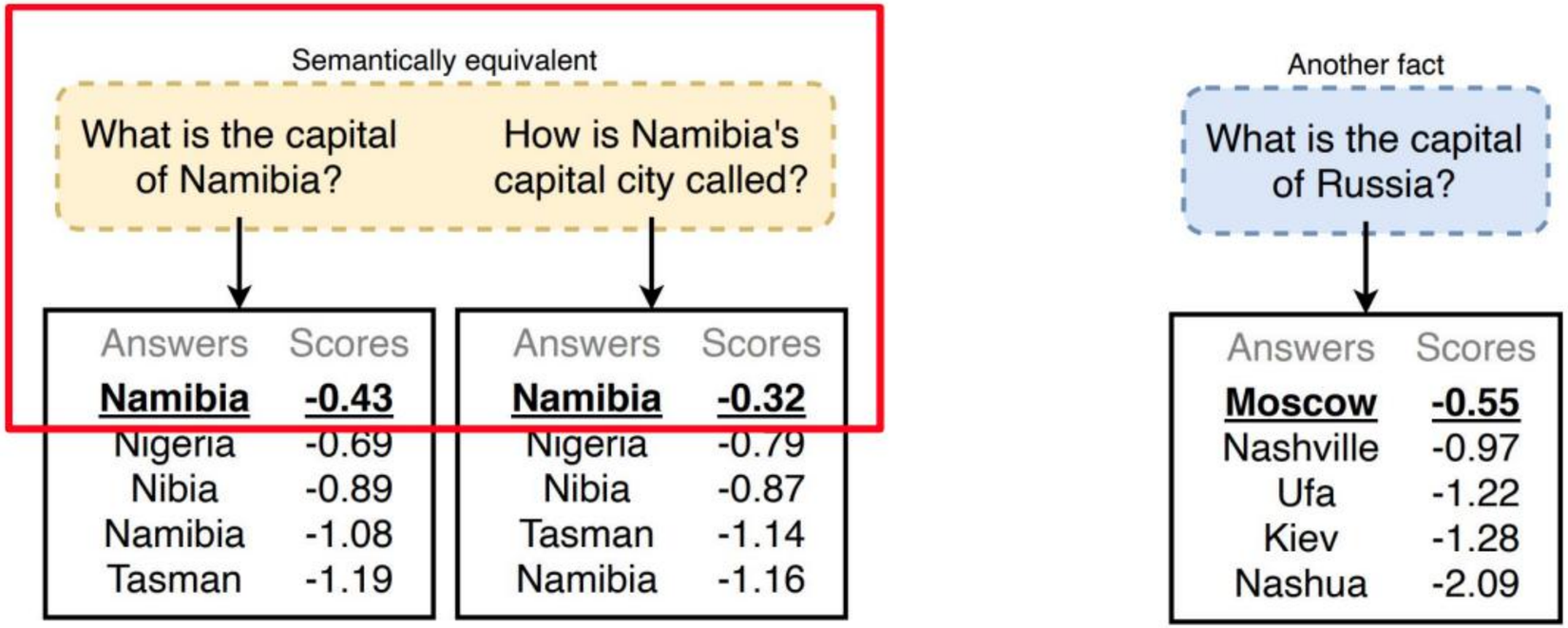
*aka **KnowledgeEditor:** https://github.com/nicola-decao/KnowledgeEditor

erc
SAP

Background

Goal 🎯

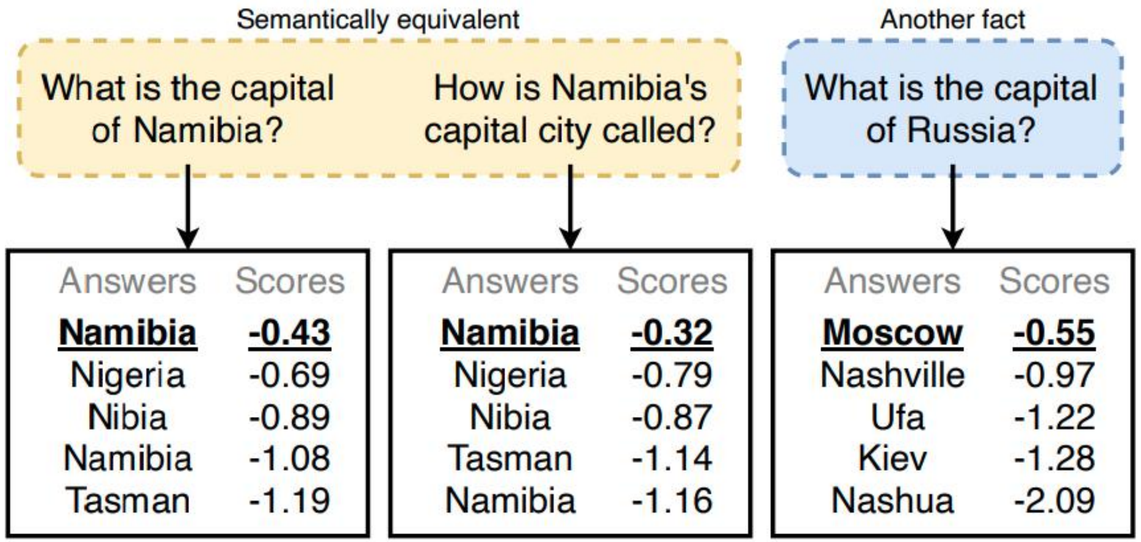
Wrong 😢 is “Windhoek”



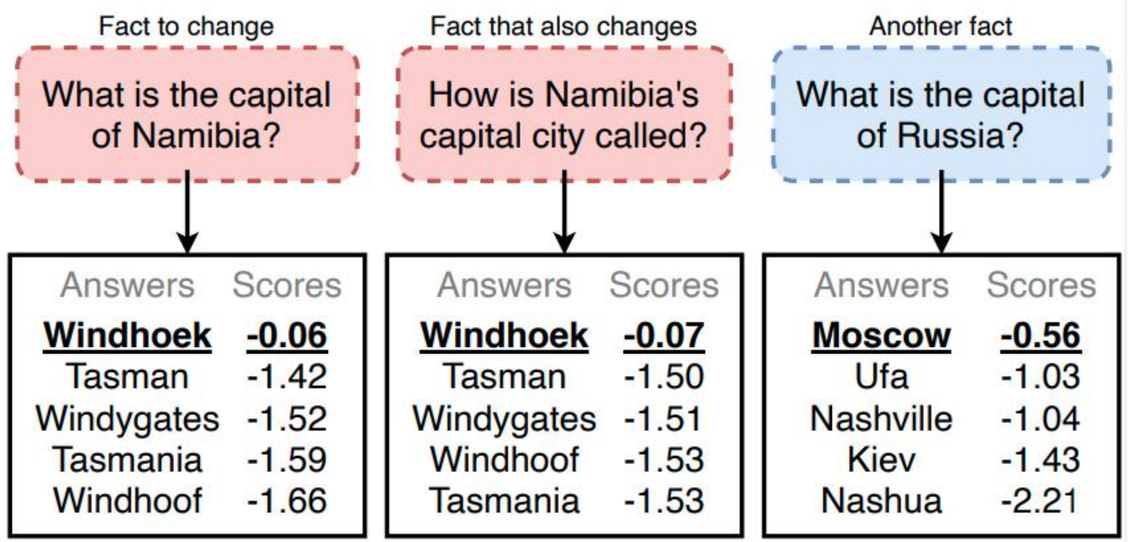
Research question: **“Can we change the factual knowledge of the model without altering all the other memorized facts?”**

Example

- BART finetuned for closed-book question answering



(a) Model predictions before the update.



(b) Model predictions with edited parameters.

Definition

- model $x \mapsto f(x; \theta)$
- revised dataset $\langle x, y, a \rangle \in \mathcal{D}$
- $\mathcal{P}^x$ semantically equivalent inputs with x
  - automatically-generated paraphrases using back-translation
- $\mathcal{O}^x$ unrelated inputs with x
- knowledge eiditor g

Methods

- Optimization

$$\min_{\phi} \quad \sum_{\hat{x} \in \mathcal{P}^x} \mathcal{L}(\theta'; \hat{x}, a)$$

$$\text{s.t.} \quad \mathcal{C}(\theta, \theta', f; \mathcal{O}^x) \leq m \quad \Rightarrow \quad \min_{\phi} \max_{\alpha} \; f(x, \theta) + \alpha \cdot (\mathcal{C}(y, \theta) - m)$$

$$\theta' = \theta + g(x, y, a; \phi)$$

- Constraint

$$\mathcal{C}_{KL}(\theta, \theta', f; \mathcal{O}^x) = \sum_{x' \in \mathcal{O}^x} \sum_{c \in \mathcal{Y}} p_{Y|X}(c|x', \theta) \log \frac{p_{Y|X}(c|x', \theta)}{p_{Y|X}(c|x', \theta')}$$

$$\mathcal{C}_{L_p}(\theta, \theta', f; \mathcal{O}^x) = \left(\sum_i |\theta_i - \theta'_i|^p\right)^{1/p}$$

Methods

- architecture for knowledge eiditor

$$g(x, y, a; \phi) \longrightarrow \Delta\theta$$

- naive hyper-network might be over-parameterized ✖
- for weight matrix $W^{n \times m} \in \theta$

$$\Delta W = \sigma(\eta) \cdot \left(\hat{\alpha} \odot \nabla_W \mathcal{L}(W; x, a) + \hat{\beta} \right)$$

$$\text{with} \quad \hat{\alpha} = \hat{\sigma}(\alpha)\gamma^\top \quad \text{and} \quad \hat{\beta} = \hat{\sigma}(\beta)\delta^\top$$

σ Sigmoid

$\hat{\sigma}$ Softmax

$$\langle x, y, a \rangle \xrightarrow[\text{Bi-LSTM}]{} h \xrightarrow[\text{5 MLP}]{} \alpha, \beta \in \mathbb{R}^m, \gamma, \delta \in \mathbb{R}^n \text{ and a scalar } \eta \in \mathbb{R}$$

Evaluation

- success rate $\mathcal{D}$
- retain accuracy $\mathcal{O}^x$
- equivalence accuracy $\mathcal{P}^x$
- performance deterioration $1 - \frac{\text{accuracy of } f(\cdot;\theta')}{\text{accuracy of } f(\cdot;\theta)}$

Experiments

- Binary FEVER dataset-> closed-book fact-checking
  - Bert model
  - flipping the predicted labels
- Zero-Shot Relation Extraction (zsRE) dataset ->closed-book QA
  - Bart model
  - pick all hypotheses enumerated via beam search except the top-1

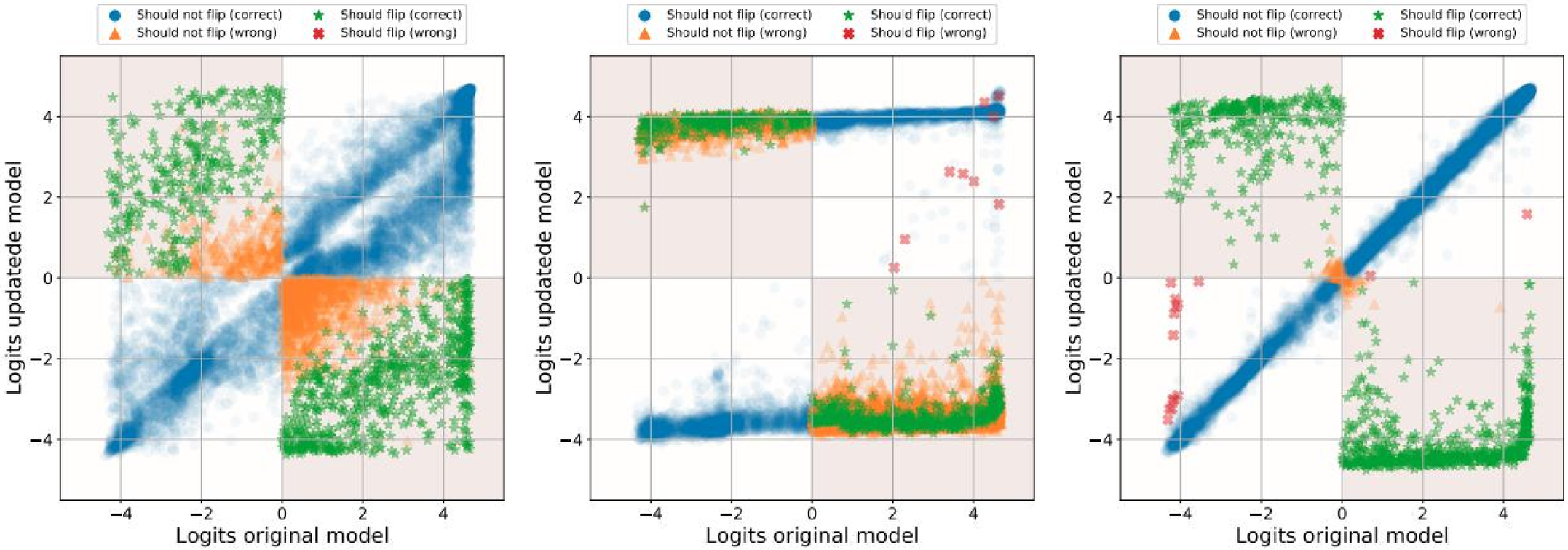
Experiments

Method	Fact-Checking				Question Answering			
	Success rate ↑	Retain acc ↑	Equiv. acc ↑	Perform. det ↓	Success rate ↑	Retain acc ↑	Equiv. acc ↑*	Perform. det ↓
Fine-tune (1st layer)	100.0	99.44	42.24	0.00	98.68	91.43	89.86 / 93.59	0.41
Fine-tune (all layers)	100.0	86.95	95.58	2.25	100.0	67.55	97.77 / 98.84	4.50
Zhu et al. (1st layer)	100.0	99.44	40.30	0.00	81.44	92.86	72.63 / 78.21	0.32
Zhu et al. (all layers)	100.0	94.07	83.30	0.10	80.65	95.56	76.41 / 79.38	0.35
Ours $\mathcal{C}_{L_2}$	99.10	45.10	99.01	35.29	99.10	46.66	97.16 / 99.24	9.22
KNOWLEDGEEDITOR	98.80	98.14	82.69	0.10	94.65	98.73	86.50 / 92.06	0.11
+ loop†	100.0	97.78	81.57	0.59	99.23	97.79	89.51 / 96.81	0.50
+ $\mathcal{P}^x$ ‡	98.50	98.55	95.25	0.24	94.12	98.56	91.20 / 94.53	0.17
+ $\mathcal{P}^x$ + loop‡	100.0	98.46	94.65	0.47	99.55	97.68	93.46 / 97.10	0.95

fine-tuning with an L_∞ constraint

generated/human annotated

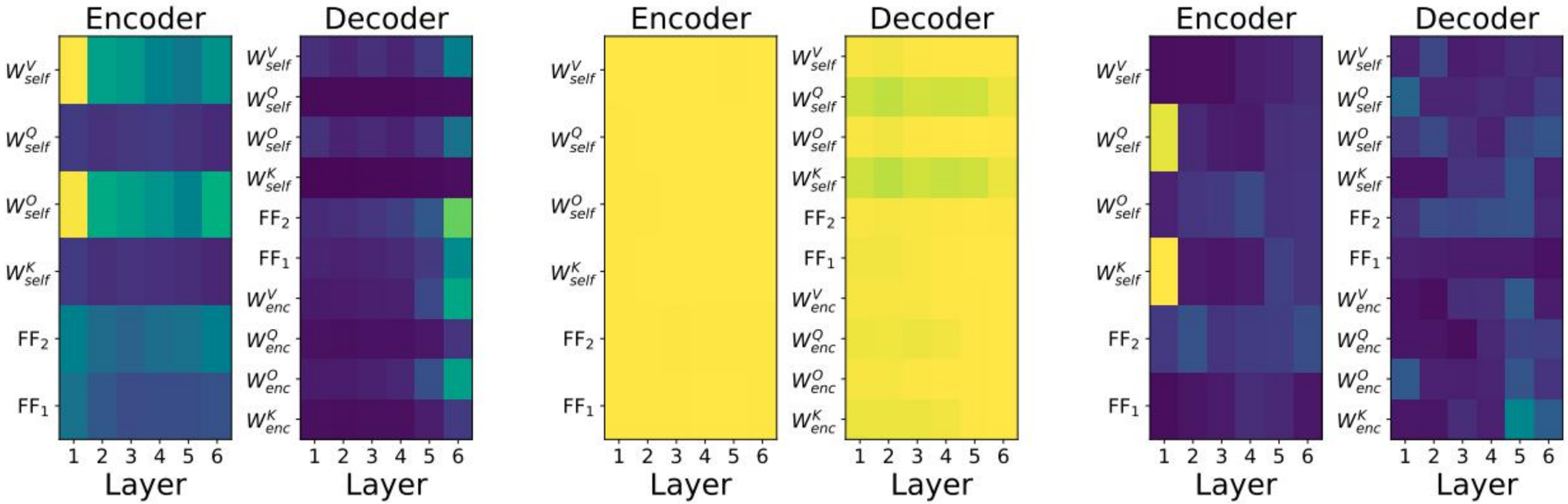
Experiments



(a) Fine-tune (all layers). (b) $\mathcal{C}_{L_2}$. (c) Ours $\mathcal{C}_{KL}$ with $\mathcal{P}^x$.

Distribution of logits of the original model and updated model on FEVER.

Experiments



(a) Gradients. (b) Fine-tune (all layers). (c) KNOWLEDGEEDITOR + $\mathcal{P}^x$.

Figure 4: Average normalized magnitude of updates on weight matrices across layers for the QA experiment. Fine-tuning updates all layers uniformly while our updates are more sparse.

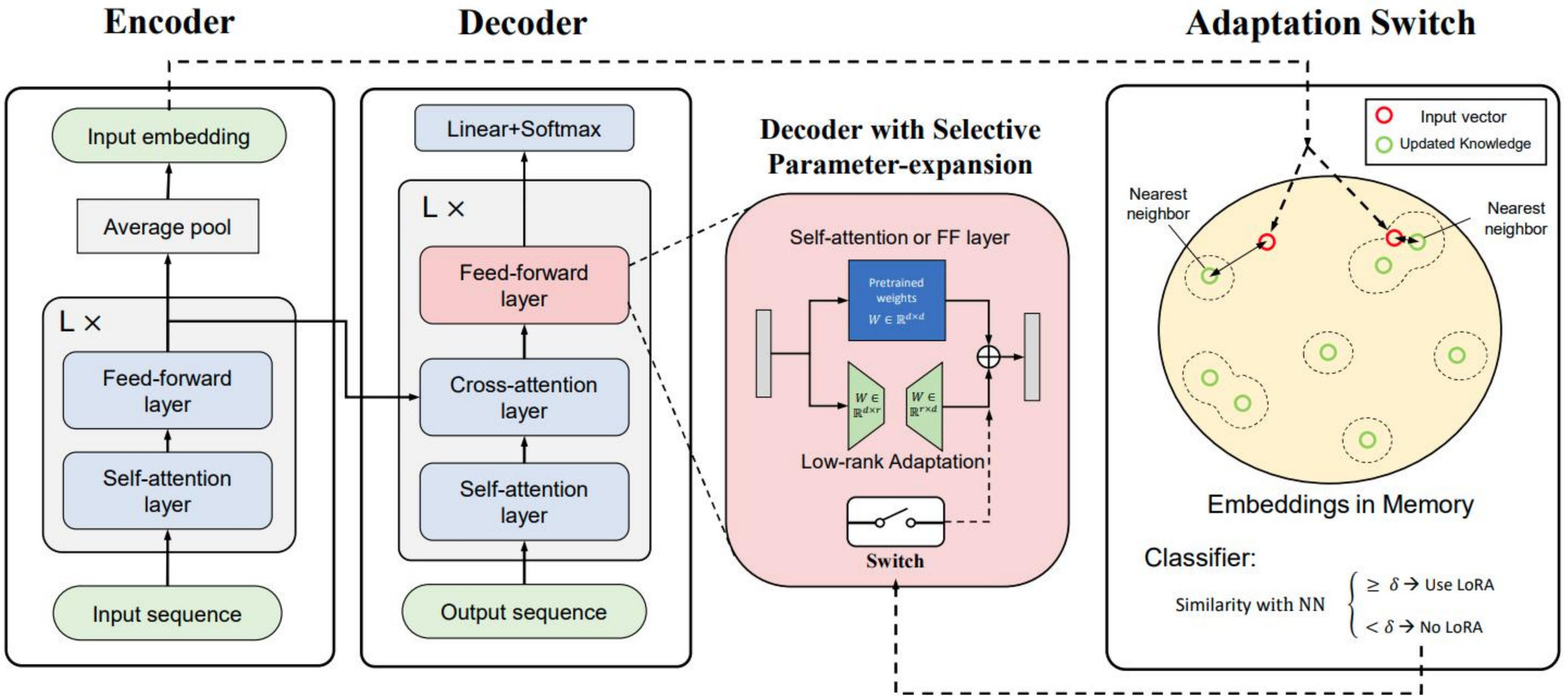
the hyper-network may be regarded as a probe providing insights about the mechanism used by the model to encode the knowledge

Plug-and-Play Adaptation for Continuously-updated QA

Kyungjae Lee[5] **Wookje Han**[1] **Seung-won Hwang**[1*]
Hwaran Lee[2] **Joonsuk Park**[2,4] **Sang-Woo Lee**[2,3]
[1]Seoul National University [2]NAVER AI Lab [3]NAVER CLOVA
[4]University of Richmond [5]LG AI Research

Findings of ACL 2022

Methods


Encoder
Decoder
Adaptation Switch
Input embedding
Average pool
L ×
Feed-forward layer
Self-attention layer
Input sequence
Linear+Softmax
L ×
Feed-forward layer
Cross-attention layer
Self-attention layer
Output sequence
Decoder with Selective Parameter-expansion
Self-attention or FF layer
Pretrained weights
$W \in \mathbb{R}^{d \times d}$
$W \in \mathbb{R}^{d \times r}$
$W \in \mathbb{R}^{r \times d}$
Low-rank Adaptation
Switch
Input vector
Updated Knowledge
Nearest neighbor
Nearest neighbor
Embeddings in Memory
Classifier:
Similarity with NN
$\geq \delta \rightarrow$ Use LoRA
$< \delta \rightarrow$ No LoRA

Methods

- selector: key(m_i)-value(g) lookup

$$h = f(x) + \sigma(q) \cdot g(x)$$

- memory embeddings $\mathcal{M} \in \mathrm{R}^{N \times d}$

$$s_q = \max_i(\mathrm{sim}(m_i, q)), \quad m_i \in \mathcal{M}$$

average the hidden states

$$\sigma(q) = \begin{cases} 1 & \text{if } s_q \geq \delta, \\ 0 & \text{if } s_q < \delta. \end{cases}$$

Methods

- multiple knowledge updates

$$h = f(x) + \sum_{k=1}^{M} \sigma_k(q) \cdot g_k(x)$$

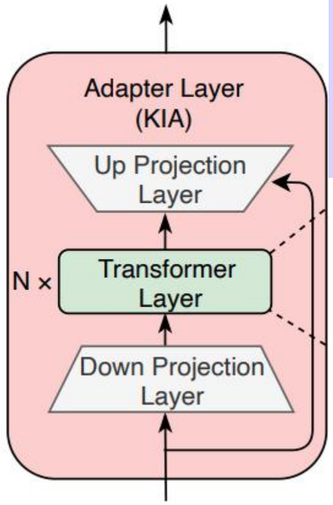
$$m^* = \underset{m}{\operatorname{argmax}}(\operatorname{sim}(m, q)), \quad m \in \mathcal{M}_{1:M}$$

$$\sigma_k(q) = \begin{cases} 1 & \text{if } s_q \geq \delta \text{ and } 1 \leq k \leq j, \\ 0 & \text{if } s_q < \delta. \end{cases} \qquad m^* \in \mathcal{M}_j$$

Experiment

- Zero-shot Relation Extraction (zsRE)
  - split into $\mathcal{K}_s$ and $\mathcal{K}_t$
- Natural Questions (NQ) + SituatedQA
  - back-translation to generate paraphrases

Experiment



	Method	# of Prams (train/total)	zsRE Question Answering $\mathcal{K}_s$	$\mathcal{P}_s$	$\mathcal{K}_t$	$\mathcal{P}_t$	F/U Ratio	NQ (with SituatedQA) $\mathcal{K}_s$	$\mathcal{P}_s$	$\mathcal{K}_t$	$\mathcal{P}_t$	F/U Ratio
	Model θ^{old}	-	95.6	95.2	25.7	28.5	-	96.6	94.9	35.3	33.7	-
	B-I: Fine-tuning	737M / 737M	76.7	70.6	92.6	85.9	0.284	92.9	82.5	94.9	**92.9**	0.435
L_p norm	B-II: RecAdam	737M / 737M	80.5	74.7	91.6	83.5	0.230	93.1	82.1	93.8	92.1	0.419
	B-III: K-adapter	538M / 840M	80.5	70.8	**96.4**	89.6	0.215	94.4	81.4	94.8	89.4	0.259
	B-IV: LoRA	62M / 799M	71.1	62.9	92.9	84.8	0.366	89.8	74.0	94.0	90.5	0.800
	Ours (+K-adapter)	538M / 840M	86.3	78.9	**96.4**	**91.1**	0.132	**95.6**	88.1	94.9	90.3	0.118
	Ours (+LoRA)	62M / 799M	**90.5**	**90.6**	95.3	89.4	**0.073**	**95.6**	**95.2**	**95.1**	90.0	**0.117**

Table 3: The comparison of the continual learning results on zsRE (Large) and NQ datasets. We measure the accuracies on the knowledge $\mathcal{K}_s, \mathcal{K}_t$, and the paraphrase knowledge $\mathcal{P}_s, \mathcal{P}_t$, with the F/U ratio.

Experiment

- selector successfully classifies 88.9% of examples, while 11.1% failed.

		Ground-truth	
		Source	Target
Selector Prediction	Source	19527 (40.7%)	854 (1.8%)
	Target	4473 (9.3%)	23146 (48.2%)

The confusion matrix of Selector.

		Ground-truth	
		Source	Target
Selector Prediction	Source	95.3	35.1
	Target	70.8 (0.0)	91.7 (97.4)

The accuracies of Ours/Retrieval in four cases.

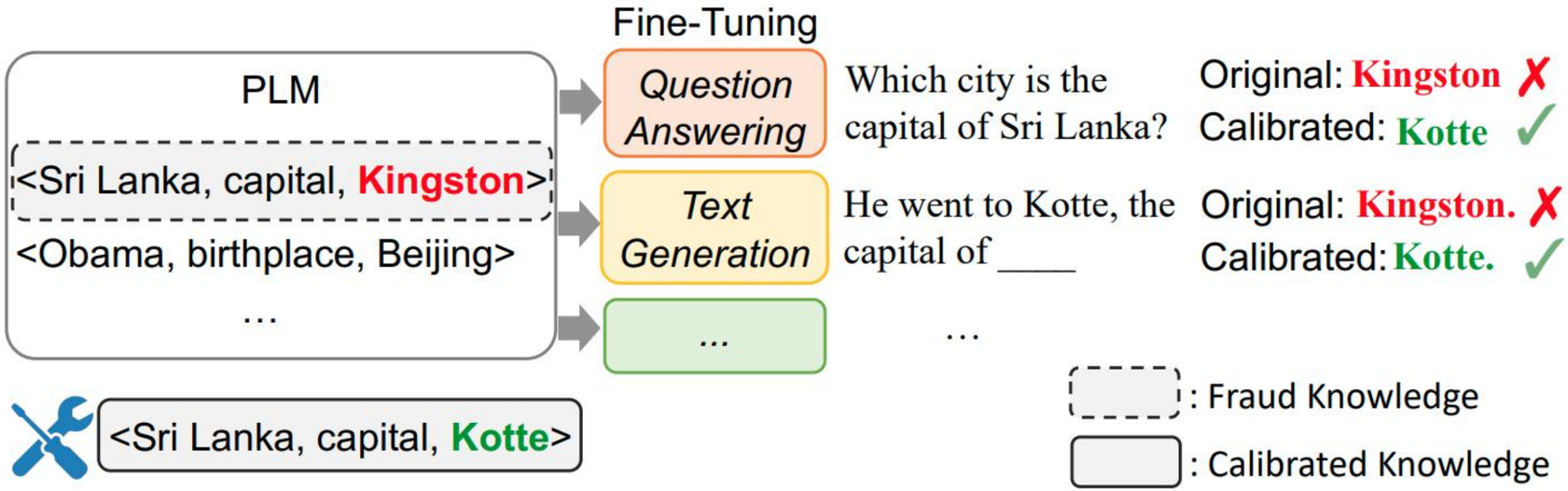
Calibrating Factual Knowledge in Pretrained Language Models

Qingxiu Dong[1] *, **Damai Dai**[1] *, **Yifan Song**[1], **Jingjing Xu**[2], **Zhifang Sui**[1] **and Lei Li**[3]
[1] Key Laboratory of Computational Linguistics, Peking University, MOE, China
[2] Shanghai AI Lab [3] University of California, Santa Barbara
dqx@stu.pku.edu.cn, {daidamai,yfsong,szf}@.pku.edu.cn,
jingjingxu@pku.edu.cn, lilei@cs.ucsb.edu

Findings of EMNLP 2022

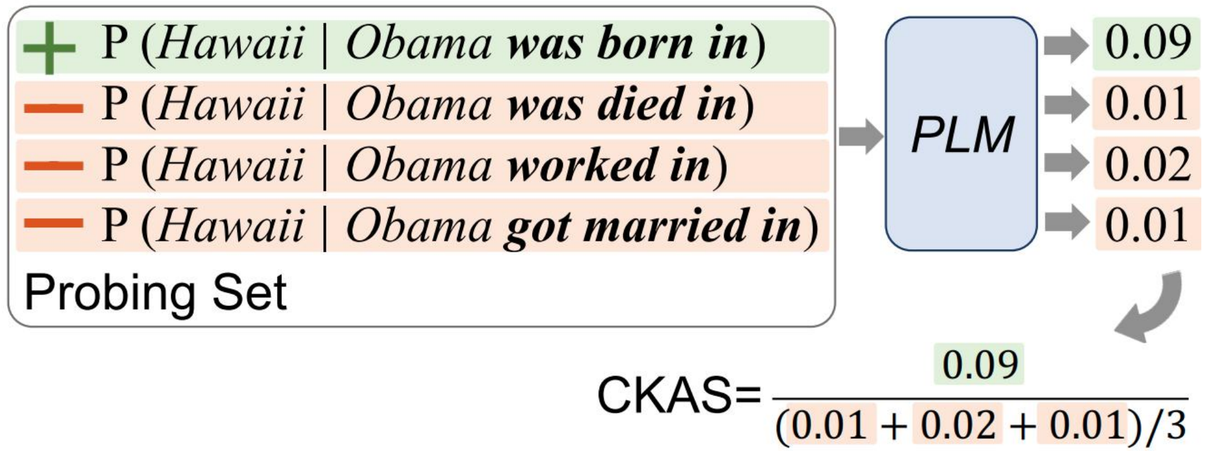
Motivation

- PLMs have factual errors



Methods

- detect wrong facts
- rank-based metric
  - inexhaustible answers
  - frequency bias
- Contrastive Knowledge Assessment



$$\mathrm{CKA_M}(s,r,o) = \frac{P_M(o|s,r) + \alpha}{\mathbb{E}_{r'}\left[P_M(o|s,r')\right] + \alpha}$$

threshold (usually < 1.0) for false facts

Methods

- detect wrong facts

Fact	Rank-based Assessment		CKA	
	Assess	Top-3 Prediction	Assess	Score
Inexhaustible Answers				
Germany shares border with *Czech Republic*.	✗	France, Russia, Austria	✓	4.45
India is a member of *UN*.	✗	NATO, India, AS	✓	2.27
Frederick was born in *Berlin*.	✗	Frederick, 18, Baltimore	✓	3.52
Frequency Bias				
Adi Shankara is affiliated with the *Hindu* religion.	✓	Hindu, Ko, Si	✗	0.98
Adi Shankara is against the *Hindu* religion.	-	Hindu, religion, Buddhist	-	-

Methods

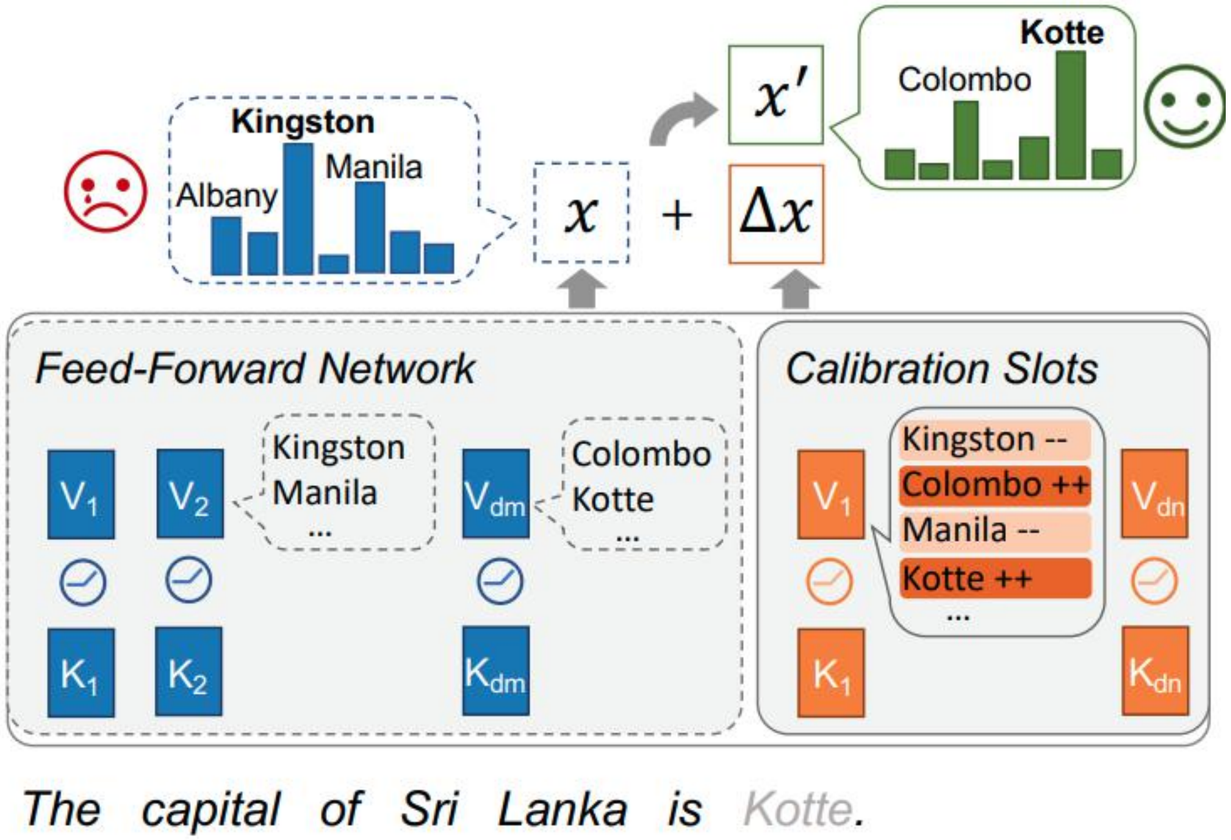
- Knowledge Calibration

$$\mathrm{FFN}(H) = \mathrm{GELU}\left(HK^T\right)V, \quad K, V \in \mathbb{R}^{d_m \times d}$$

$$\Delta\mathrm{FFN}(H) = \mathrm{GELU}\left(H\tilde{K}^T\right)\tilde{V},$$

$$\mathrm{FFN}'(H) = \mathrm{FFN}(H) + \Delta\mathrm{FFN}(H),$$

$$\tilde{K}, \tilde{V} \in \mathbb{R}^{d_c \times d} \qquad d_c \ll d_m$$



Data& Model Setup

- factual triplets from the T-REx dataset
- fill head entity and tail entity into the template in LAMA
- 100/1000 facts
- write 3 erroneous relation templates for each relation in LAMA

Data& Model Setup

- PARAREL dataset
  - contains various surface form templates for 38 relations
- masked language modeling objective on calibration data

Split	Source	Target
Train	[MASK] was born in Hawaii.	Obama
	Obama is originally from [MASK].	Hawaii
	[MASK] was originally from Hawaii.	Obama
	Obama is native to [MASK].	Hawaii
Valid	[MASK] originates from Hawaii.	Obama
	Obama originated from [MASK].	Hawaii
Test	Obama is a/an [MASK]-born person.	Hawaii
	[MASK] was native to Hawaii.	Obama
	Obama, a [MASK]-born person.	Hawaii

Table 2: Example of knowledge-intensive data for training CALINET. We generate multiple texts via templates for each triple where the templates in training, validation, and test are not sharing.

Experiment

target entity is replaced by a false one in the same entity type.

randomly mask the test data as pretraining

continue pretraining method

Model	# Facts	Method	# Calibration Params	False Rate(↓)	Ori (↓)	Adv (↑)	LM(↓)	EM(↑)	F1(↑)
T5-base	10^2	Vanilla	0	48.10%	87.21	219.18	89.21	0.63	7.48
		CALINET	0.1M	17.09%	1.22	>1000	54.45	81.65	84.58
		C. P.	220M	13.29%	1.15	>1000	116.52	87.34	89.85
	10^3	Vanilla	0	51.34%	90.61	208.90	60.64	0.94	6.51
		CALINET	0.5M	18.30%	1.26	>1000	46.71	71.18	73.48
		C. P.	220M	18.23%	1.28	>1000	139.96	78.15	80.35
T5-large	10^2	Vanilla	0	46.20%	34.36	116.38	92.52	2.53	7.23
		CALINET	0.5M	15.19%	1.30	>1000	44.21	81.65	85.11
		C. P.	770M	14.56%	1.21	>1000	477.24	87.97	90.49
	10^3	Vanilla	0	45.04%	31.44	93.77	58.78	2.48	6.86
		CALINET	1.0M	20.84%	1.32	>1000	43.04	70.84	72.92
		C. P.	770M	17.16%	1.28	>1000	154.52	78.22	80.57

"Ori." and "Adv." refer to model perplexity on the original test set (contains true facts) and the adversarial test set (contains false facts), respectively. We concatenate CALINET to the last layer of the T5 decoder.

Experiment

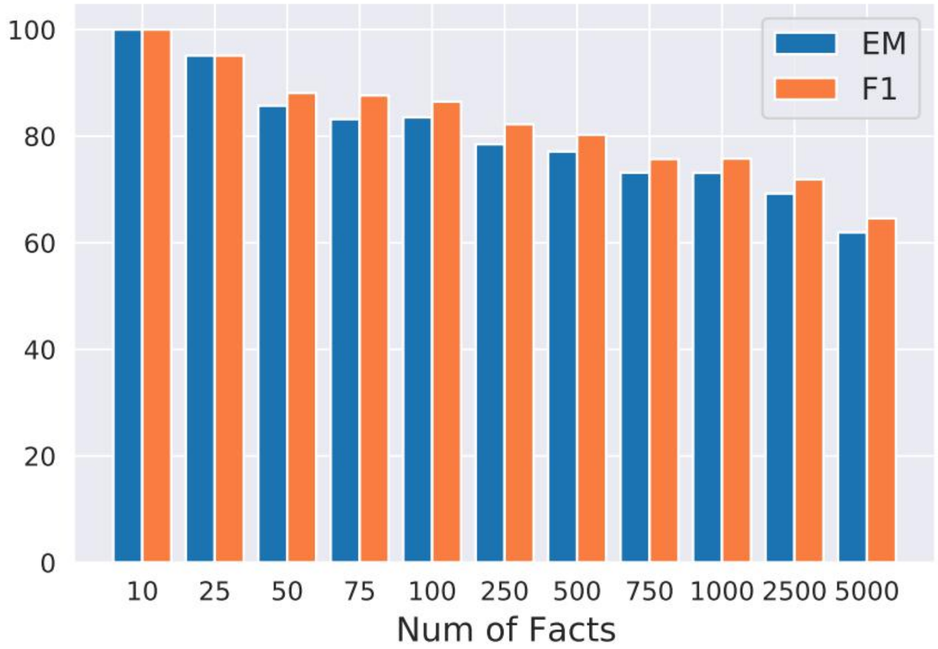


Figure 4: Calibration results for different scales of facts. Given 5000 facts, our method can calibrate more than 60% of facts in PLMs at once.

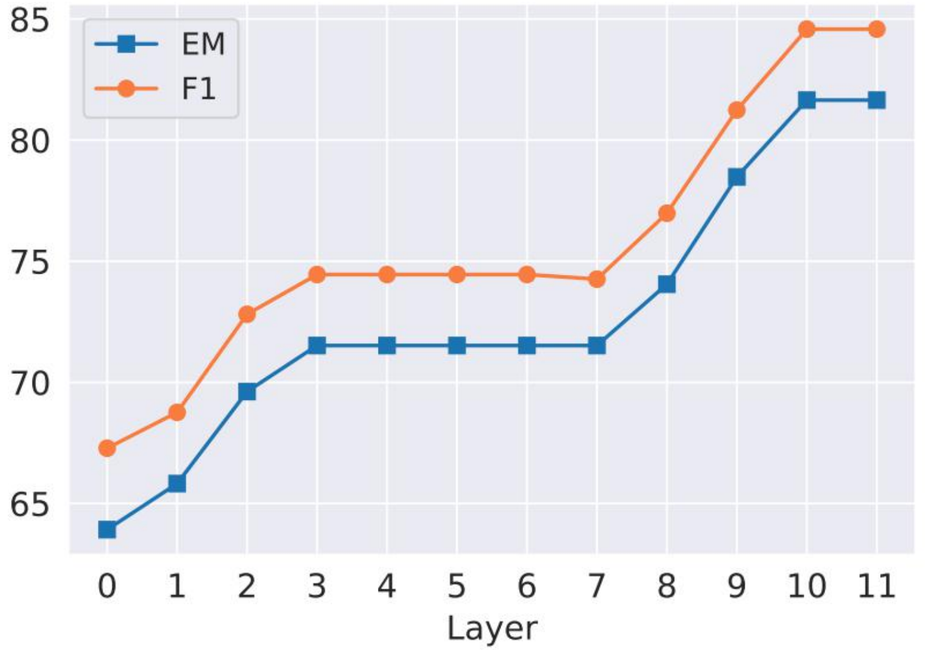


Figure 5: Calibration ability of concatenating CaliNet in different layers.

Repairing Neural Networks by Leaving the Right Past Behind

Ryutaro Tanno
Microsoft Research Cambridge, UK
rytanno@microsoft.com

Melanie F. Pradier
Microsoft Research Cambridge, UK
melanief@microsoft.com

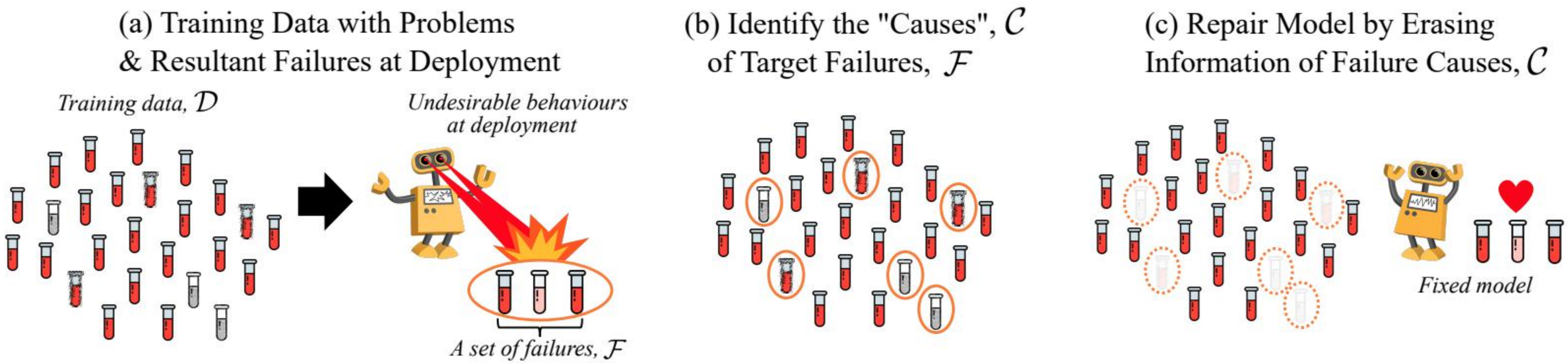
Aditya Nori
Microsoft Research Cambridge, UK
Aditya.Nori@microsoft.com

Yingzhen Li
Imperial College London, UK
yingzhen.li@imperial.ac.uk

NeurIPS 2022

Motivation

- annotation noise, low-quality inputs, anomalies, and acquisition biases
- finding & erasing



Model Repairment by Data Deletion

- **Cause identification**: Identify a set of detrimental datapoints C in the training data D that contributed the most to the failure set F.

- **Treatment**: Given the set of failure causes C, adapt the model to predict correctly on the failure set F, while maintaining performance on remaining test examples. -> unlearn C

Cause identification

- Given D, F, finding C ⊆ D

$$r(\mathcal{C}) := \log p(\mathcal{F}|\mathcal{D}\setminus\mathcal{C}) - \log p(\mathcal{F}|\mathcal{D}) \qquad \mathcal{C} = \operatorname{argmax}_{\mathcal{C}'\in\mathbb{P}(\mathcal{D})} r(\mathcal{C}')$$

unacceptable

$$\begin{aligned}
\log p(\mathcal{F}|\mathcal{D}\setminus\mathcal{C}) &= \log \int p(\mathcal{F}|\boldsymbol{\theta}) p(\boldsymbol{\theta}|\mathcal{D}\setminus\mathcal{C}) d\boldsymbol{\theta} \\
&= \log \int p(\mathcal{F}|\boldsymbol{\theta}) \frac{p(\mathcal{D}\setminus\mathcal{C}|\boldsymbol{\theta}) p(\boldsymbol{\theta})}{p(\mathcal{D}\setminus\mathcal{C})} d\boldsymbol{\theta} && \text{(Bayes' rule)} \\
&= \log \int p(\mathcal{F}|\boldsymbol{\theta}) \frac{p(\mathcal{D}|\boldsymbol{\theta}) p(\boldsymbol{\theta})}{p(\mathcal{C}|\boldsymbol{\theta}) p(\mathcal{D}\setminus\mathcal{C})} d\boldsymbol{\theta} && \text{(by Eq. (18))} \\
&= \log \int \frac{p(\mathcal{D})}{p(\mathcal{D}\setminus\mathcal{C})} \cdot \frac{p(\mathcal{F}|\boldsymbol{\theta})}{p(\mathcal{C}|\boldsymbol{\theta})} \cdot \frac{p(\mathcal{D},\boldsymbol{\theta})}{p(\mathcal{D})} d\boldsymbol{\theta} && \text{(multiplying } \frac{p(\mathcal{D})}{p(\mathcal{D})} \text{ and rearranging terms)} \\
&= \log \int \frac{p(\mathcal{F}|\boldsymbol{\theta})}{p(\mathcal{C}|\boldsymbol{\theta})} p(\boldsymbol{\theta}|\mathcal{D}) d\boldsymbol{\theta} + \log \frac{p(\mathcal{D})}{p(\mathcal{D}\setminus\mathcal{C})}. && \text{(Bayes' rule)}
\end{aligned}$$

Cause identification

$$r(\mathcal{C}) := \log p(\mathcal{F}|\mathcal{D}\setminus\mathcal{C}) - \log p(\mathcal{F}|\mathcal{D})$$

$$\log p(\mathcal{F}|\mathcal{D}\setminus\mathcal{C}) = \log \int \frac{p(\mathcal{F}|\boldsymbol{\theta})}{p(\mathcal{C}|\boldsymbol{\theta})} p(\boldsymbol{\theta}|\mathcal{D}) d\boldsymbol{\theta} + \log \frac{p(\mathcal{D})}{p(\mathcal{D}\setminus\mathcal{C})}$$

$$r(\mathcal{C}) = \log \left(\frac{p(\mathcal{F}|\mathcal{D}\setminus\mathcal{C})}{p(\mathcal{F}|\mathcal{D})} \right)$$

$$= \log \int \frac{p(\mathcal{F}|\boldsymbol{\theta})}{p(\mathcal{C}|\boldsymbol{\theta})} p(\boldsymbol{\theta}|\mathcal{D}) d\boldsymbol{\theta} - \log p(\mathcal{F}|\mathcal{D}) + \log p(\mathcal{D}) - \log \int p(\mathcal{D}\setminus\mathcal{C}|\boldsymbol{\theta}) p(\boldsymbol{\theta}) d\boldsymbol{\theta}$$

(by definition of marginal distributions)

$$= \log \int \frac{1}{p(\mathcal{C}|\boldsymbol{\theta})} \frac{p(\mathcal{F}|\boldsymbol{\theta}) p(\boldsymbol{\theta}|\mathcal{D})}{p(\mathcal{F}|\mathcal{D})} d\boldsymbol{\theta} - \log \int \frac{p(\mathcal{D}|\boldsymbol{\theta})}{p(\mathcal{C}|\boldsymbol{\theta})} \frac{p(\boldsymbol{\theta})}{p(\mathcal{D})} d\boldsymbol{\theta}$$

(by Eq. (18) and rearranging terms)

$$= \log \int \frac{p(\boldsymbol{\theta}|\mathcal{D},\mathcal{F})}{p(\mathcal{C}|\boldsymbol{\theta})} d\boldsymbol{\theta} - \log \int \frac{p(\boldsymbol{\theta}|\mathcal{D})}{p(\mathcal{C}|\boldsymbol{\theta})} d\boldsymbol{\theta}. \qquad \text{(Bayes' rule)}$$

Cause identification

$$r(\mathcal{C}) = \log \int \frac{p(\boldsymbol{\theta}|\mathcal{D},\mathcal{F})}{p(\mathcal{C}|\boldsymbol{\theta})} d\boldsymbol{\theta} - \log \int \frac{p(\boldsymbol{\theta}|\mathcal{D})}{p(\mathcal{C}|\boldsymbol{\theta})} d\boldsymbol{\theta}$$

→ only need to compute p(θ|D,F) once, but still need combinatorial search for the best subset C

re-write $r(\mathcal{C}) = F(1, p(\boldsymbol{\theta}|\mathcal{D},\mathcal{F})) - F(1, p(\boldsymbol{\theta}|\mathcal{D})), \quad F(\epsilon, g(\boldsymbol{\theta})) := \log \int g(\boldsymbol{\theta}) e^{-\epsilon \log p(\mathcal{C}|\boldsymbol{\theta})} d\boldsymbol{\theta}$

Taylor expansion of $F(\epsilon, g(\boldsymbol{\theta}))$ around $\epsilon = 0$

$$F(\epsilon, g(\boldsymbol{\theta})) = -\epsilon \mathbb{E}_{g(\boldsymbol{\theta})}[\log p(\mathcal{C}|\boldsymbol{\theta})] + \mathcal{O}(\epsilon^2)$$

re-write $\hat{r}(\mathcal{C}) := \mathbb{E}_{p(\boldsymbol{\theta}|\mathcal{D})}[\log p(\mathcal{C}|\boldsymbol{\theta})] - \mathbb{E}_{p(\boldsymbol{\theta}|\mathcal{D},\mathcal{F})}[\log p(\mathcal{C}|\boldsymbol{\theta})]$

$\mathcal{O}(|\mathcal{D}|!) \longrightarrow \mathcal{O}(|\mathcal{D}|)$

data i.i.d. $\hat{r}(\boldsymbol{z}) = \mathbb{E}_{p(\boldsymbol{\theta}|\mathcal{D})}[\log p(\boldsymbol{z}|\boldsymbol{\theta})] - \mathbb{E}_{p(\boldsymbol{\theta}|\mathcal{D},\mathcal{F})}[\log p(\boldsymbol{z}|\boldsymbol{\theta})], \quad p(\boldsymbol{z}|\boldsymbol{\theta}) = p(y|\boldsymbol{x},\boldsymbol{\theta})$

Cause identification

- $p(\boldsymbol{\theta}|\mathcal{D},\mathcal{F})$ -> recomputing the posterior from scratch can be expensive
  - MLE/MAP point estimates, Laplace approximation, variational inference, etc

- Example I (Linear Influence Function)

$$\hat{\boldsymbol{\theta}}^{\star}_{\mathcal{D},\mathcal{F}} \approx \hat{\boldsymbol{\theta}} + \gamma \hat{\boldsymbol{F}}^{-1}_{\hat{\boldsymbol{\theta}}} \nabla_{\hat{\boldsymbol{\theta}}} \log p(\mathcal{F}|\hat{\boldsymbol{\theta}}) \qquad \tilde{r}(\boldsymbol{z}) = -\gamma \nabla_{\hat{\boldsymbol{\theta}}} \log p(\mathcal{F}|\hat{\boldsymbol{\theta}})^{\top} \hat{\boldsymbol{F}}^{-1}_{\hat{\boldsymbol{\theta}}} \nabla_{\hat{\boldsymbol{\theta}}} \log p(\boldsymbol{z}|\hat{\boldsymbol{\theta}})$$

- Example II (Elastic Weight Consolidation)

maximize $\log p(\mathcal{F}|\boldsymbol{\theta}) - \frac{N}{2}(\boldsymbol{\theta}-\hat{\boldsymbol{\theta}})^{\top} \hat{\boldsymbol{F}}_{\hat{\boldsymbol{\theta}}} (\boldsymbol{\theta}-\hat{\boldsymbol{\theta}}) - \frac{\lambda}{2}||\boldsymbol{\theta}-\hat{\boldsymbol{\theta}}||_2^2, \qquad \tilde{r}(\boldsymbol{z}) = \log p(\boldsymbol{z}|\hat{\boldsymbol{\theta}}) - \log p(\boldsymbol{z}|\hat{\boldsymbol{\theta}}^{\star}_{\mathcal{D},\mathcal{F}})$

Treatment

- $p(\boldsymbol{\theta}|\mathcal{D}\backslash\mathcal{C})$ -> retrain is time-consuming

- Example I: Fine-tuning on Corrected Data
- Example II: (Newton Update Removal)

$$\hat{\boldsymbol{\theta}}^{\star\star}_{\mathcal{D}\backslash\mathcal{C}} \approx \hat{\boldsymbol{\theta}} - \gamma \hat{\boldsymbol{F}}^{-1}_{\hat{\boldsymbol{\theta}}} \nabla_{\hat{\boldsymbol{\theta}}} \log p(\mathcal{C}|\hat{\boldsymbol{\theta}})$$

- Example III: (EWC for data deletion)

maximize $$-\log p(\mathcal{C}|\boldsymbol{\theta}) - \frac{N}{2}(\boldsymbol{\theta}-\hat{\boldsymbol{\theta}})^\top \hat{\boldsymbol{F}}_{\hat{\boldsymbol{\theta}}}(\boldsymbol{\theta}-\hat{\boldsymbol{\theta}}) - \frac{\lambda}{2}||\boldsymbol{\theta}-\hat{\boldsymbol{\theta}}||_2^2$$

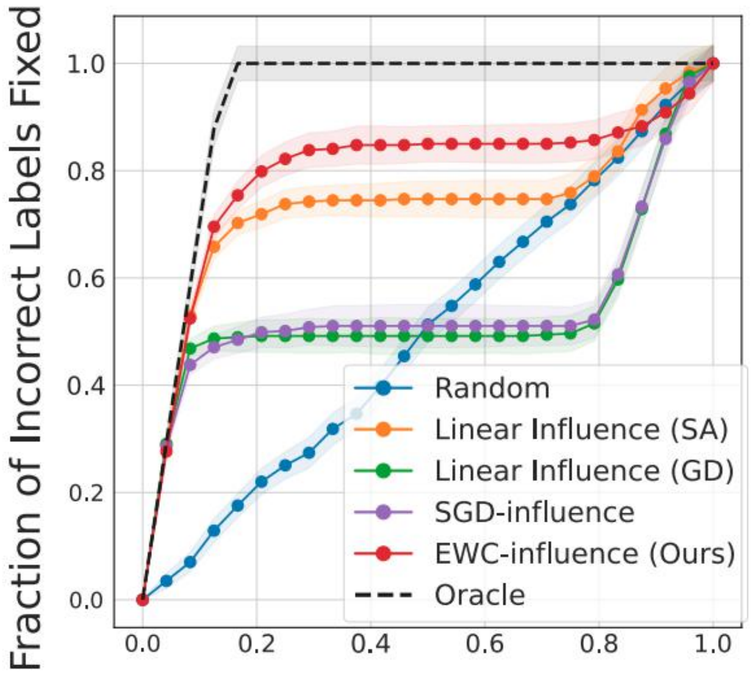
Experiment

- Dataset
  - CIFAR10
  - MNIST (only use 6% training set)

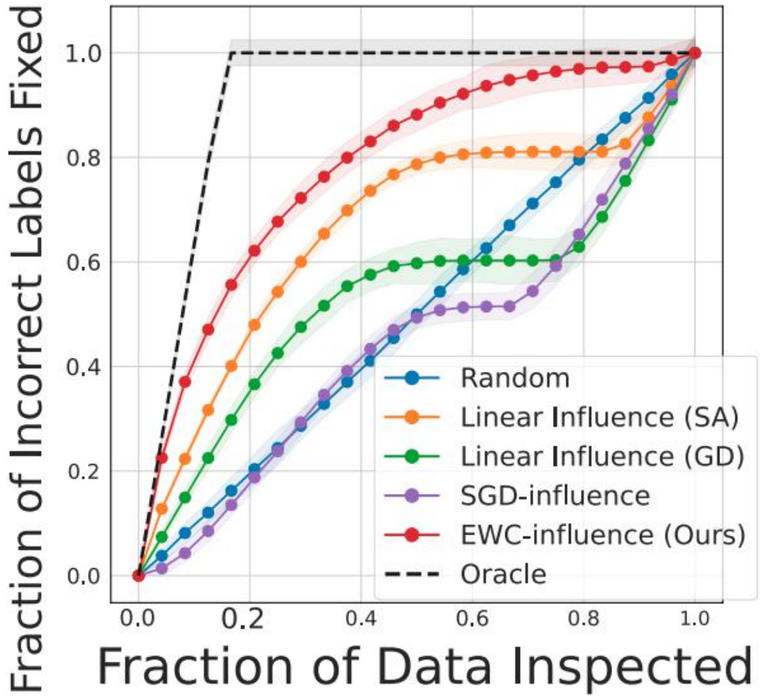
- Evaluation
  - separate the test set into F (“failure set”) and T \ F (“remaining set”)
  - split the failure set into query and holdout set, use query to identify failure causes C, use the latter to evaluate

Identifying Failure Causes

- randomly flip labels for two classes, use F_q to find incorrectly labelled samples



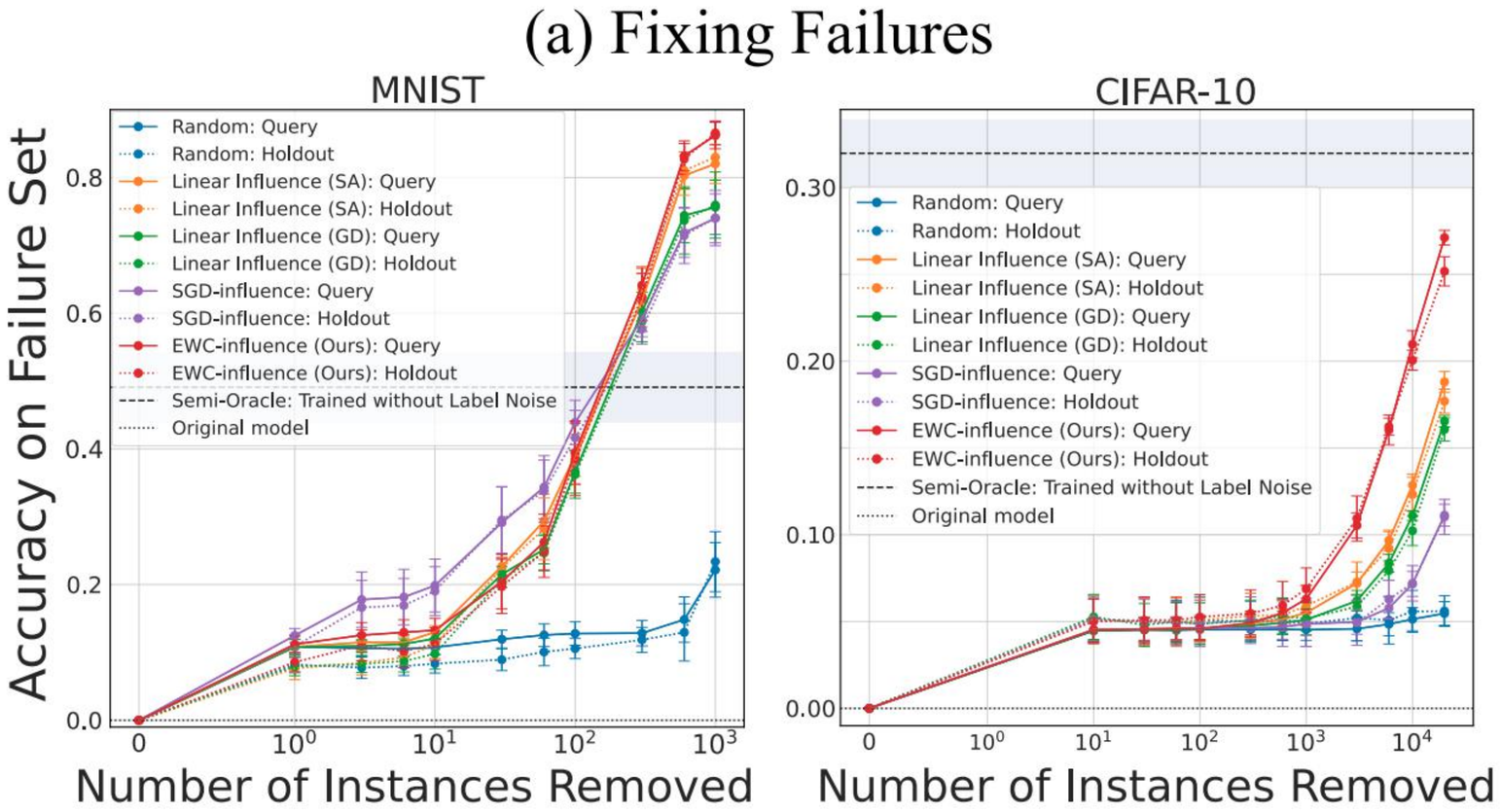
MNIST



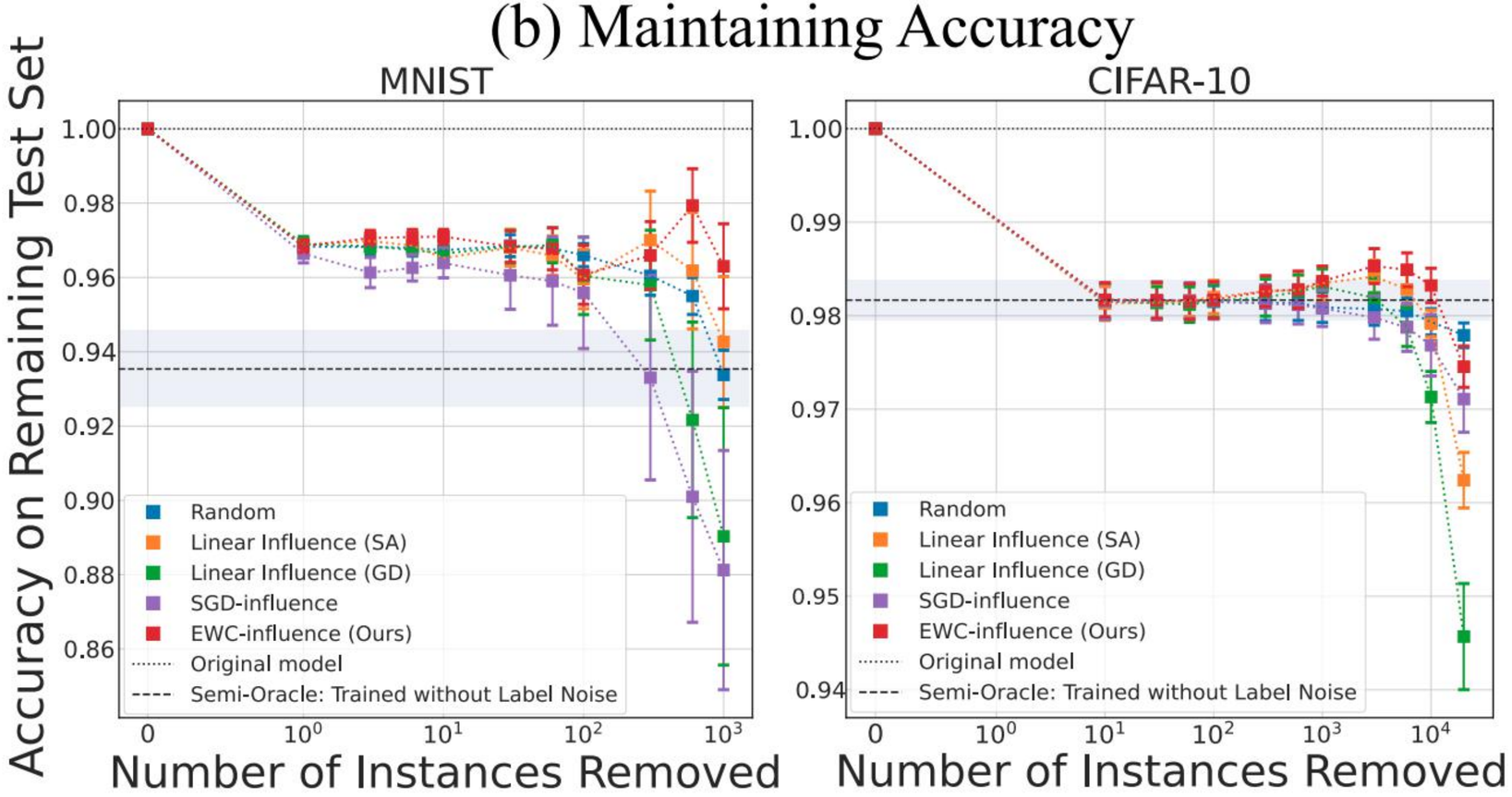
CIFAR10

Identifying Failure Causes

- fine-tune the base model on D\C

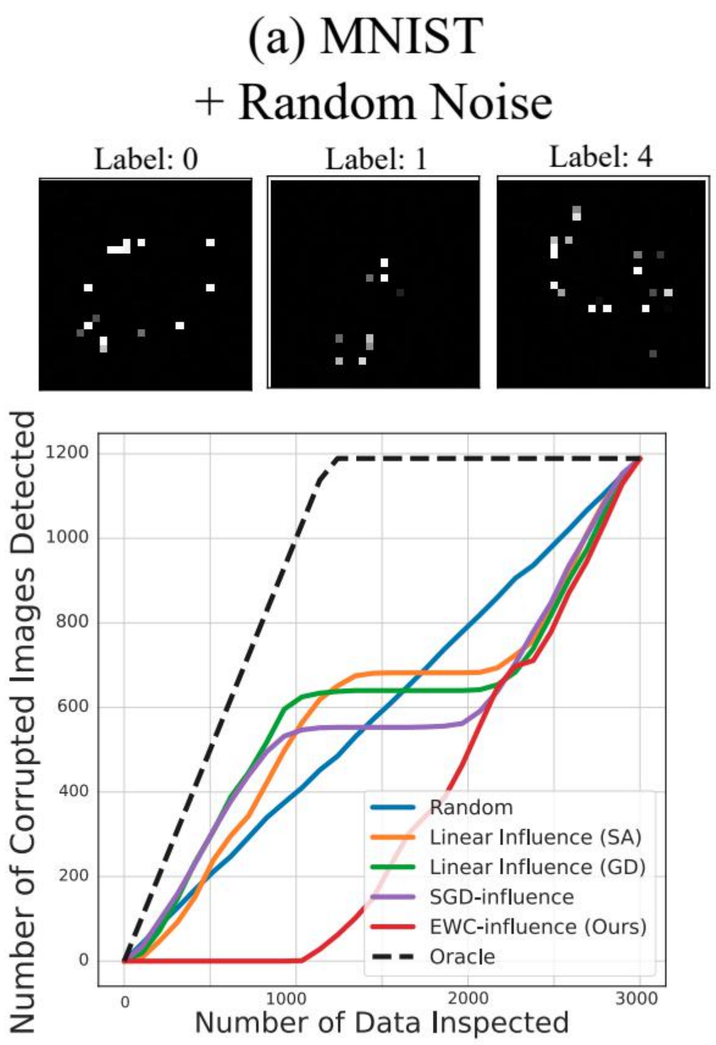


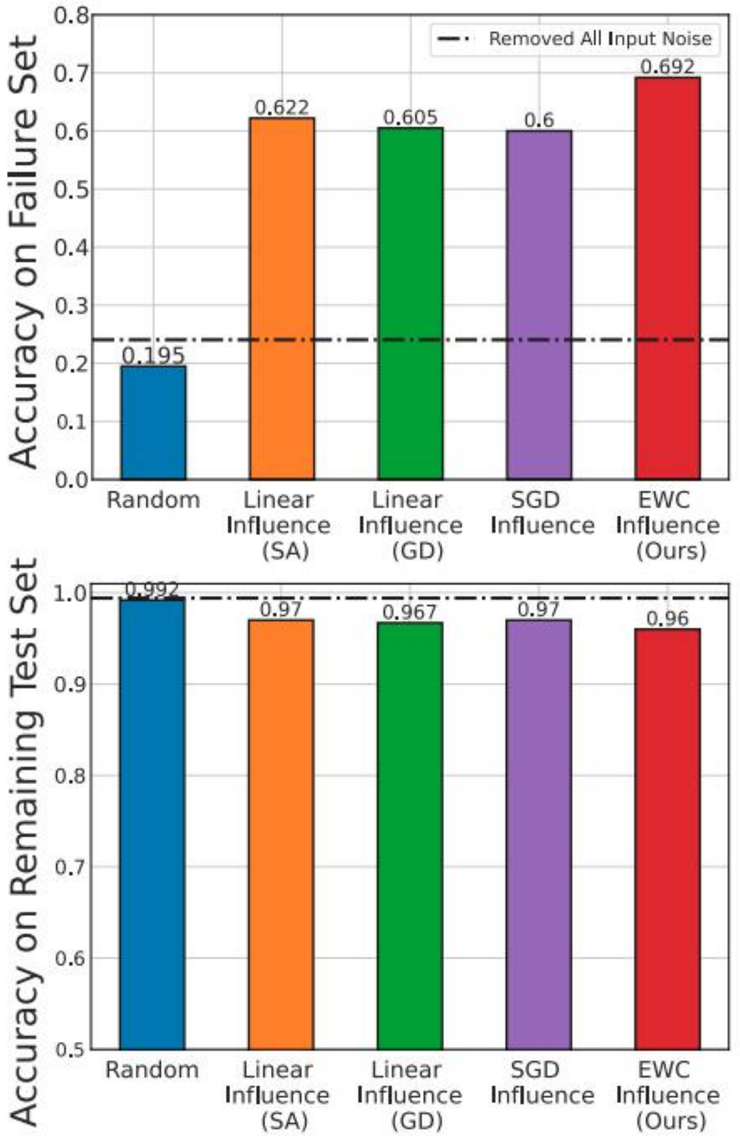
Identifying Failure Causes



Identifying Failure Causes

12%





Comparison of Treatment Methods

different
step sizes