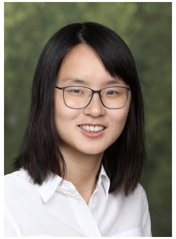
PRINCETON NLP

Factual Probing is [MASK]: Learning vs. Learning to Recall





Zexuan Zhong* Dan Friedman* Danqi Chen

Princeton University

Language Models Capture Factual Knowledge

Language Models Capture Factual Knowledge

Fact: (DirectX, developer, Microsoft)

Language Models Capture Factual Knowledge

Fact: (DirectX, developer, Microsoft)

[MASK] released the DirectX ⟹  ⟹ Microsoft

Petroni et al., 2019. Language Models as Knowledge Bases?

This Work

1. How to **generate** good prompts for factual probing?
2. Can we **trust** the probing results of optimized prompts?
3. How can we better **interpret** the probing results?

This Work

1. How to **generate** good prompts for factual probing?
2. Can we **trust** the probing results of optimized prompts?
3. How can we better **interpret** the probing results?

Prompts Matter!

[MASK] released the DirectX ⇨  ⇨ Microsoft

Prompts Matter!

[MASK] released the DirectX

Microsoft

DirectX was developed by [MASK]

Intel

Jiang et al., 2020. How Can We Know What Language Models Know?

Prompts Matter!

[MASK] released the DirectX ⇨

⇨ Microsoft ✅

DirectX was developed by [MASK] ⇨

⇨ Intel ❌

Jiang et al., 2020. How Can We Know What Language Models Know?

Generating Prompts

Generating Prompts

LAMA (Petroni et al., 2019):
manually defined

[X] is [MASK] citizen

Generating Prompts

LAMA (Petroni et al., 2019):
manually defined

[X] is [MASK] citizen

LPAQA (Jiang et al., 2020):
mined & paraphrased

Wikipedia
The Free Encyclopedia

[X] is a citizen of [MASK]

Generating Prompts

LAMA (Petroni et al., 2019):
manually defined

[X] is [MASK] citizen

LPAQA (Jiang et al., 2020):
mined & paraphrased

[X] is a citizen of [MASK]

AutoPrompt (Shin et al., 2020):
discrete-token search

[X] m^3 badminton pieces internationally representing [MASK]

Generating Prompts

LAMA (Petroni et al., 2019):
manually defined

[X] is [MASK] citizen

LPAQA (Jiang et al., 2020):
mined & paraphrased

[X] is a citizen of [MASK]

AutoPrompt (Shin et al., 2020):
discrete-token search

[X] m^3 badminton pieces internationally representing [MASK]

*Why do prompts have to be a sequence of **tokens**?*

Generating Prompts

LAMA (Petroni et al., 2019):
manually defined

[X] is [MASK] citizen

Wikipedia
The Free Encyclopedia

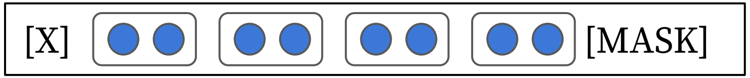
LPAQA (Jiang et al., 2020):
mined & paraphrased

[X] is a citizen of [MASK]

AutoPrompt (Shin et al., 2020):
discrete-token search

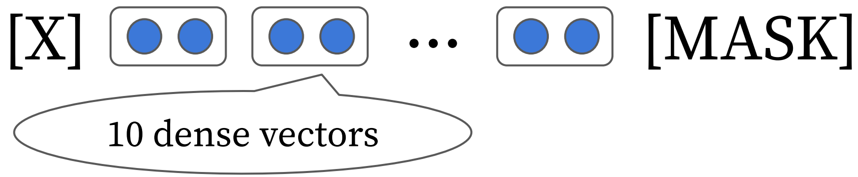
[X] m^3 badminton pieces internationally representing [MASK]

OptiPrompt (ours):
dense-vector optimization



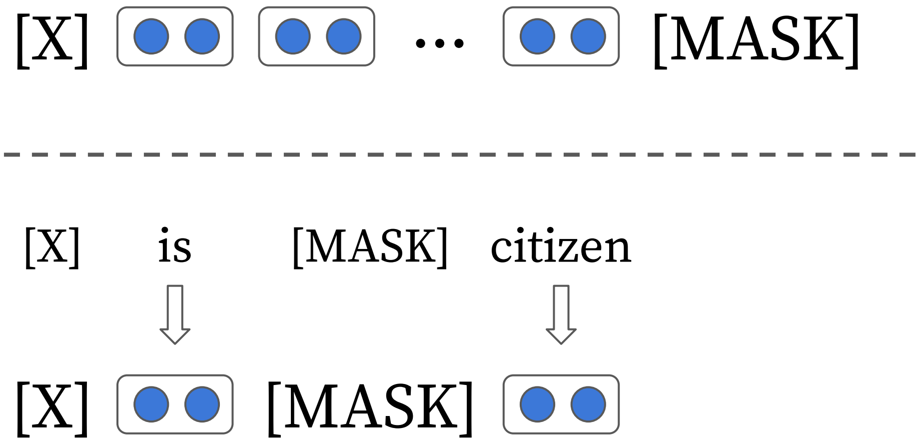
OptiPrompt

Prompt definition



OptiPrompt

Prompt definition



OptiPrompt

Prompt definition

[X] ... [MASK]

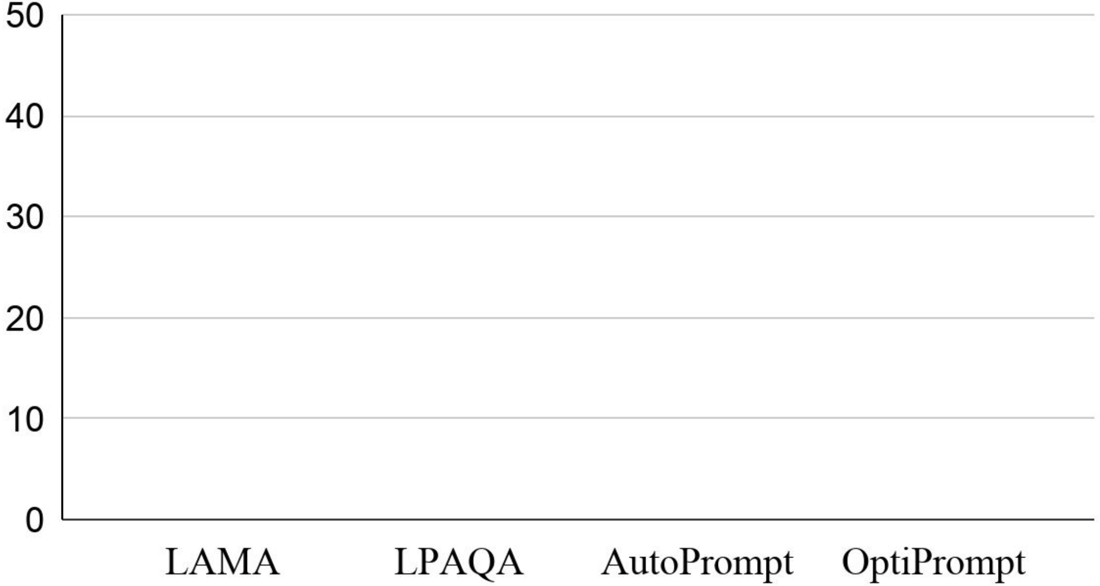
Training

$$\mathcal{L}_r = -\frac{1}{|D_r|} \sum_{(s,o)\in D_r} \log P(\texttt{[MASK]} = o \mid t_r(s))$$

1,000 (*s*, *o*) pairs for each relation *r*

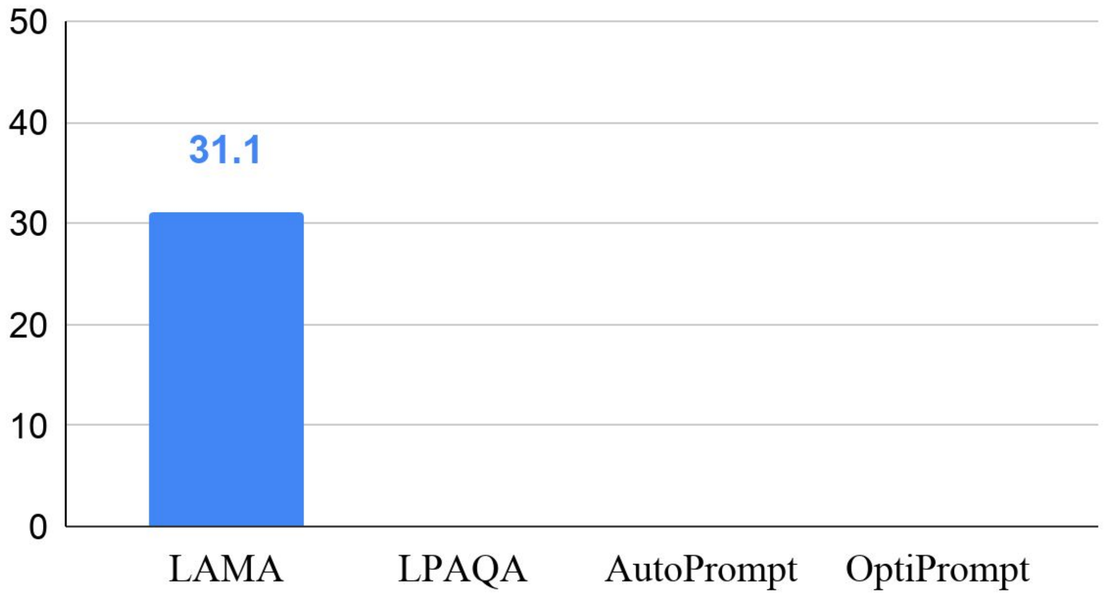
Results on the LAMA Benchmark

Results are based on BERT-base



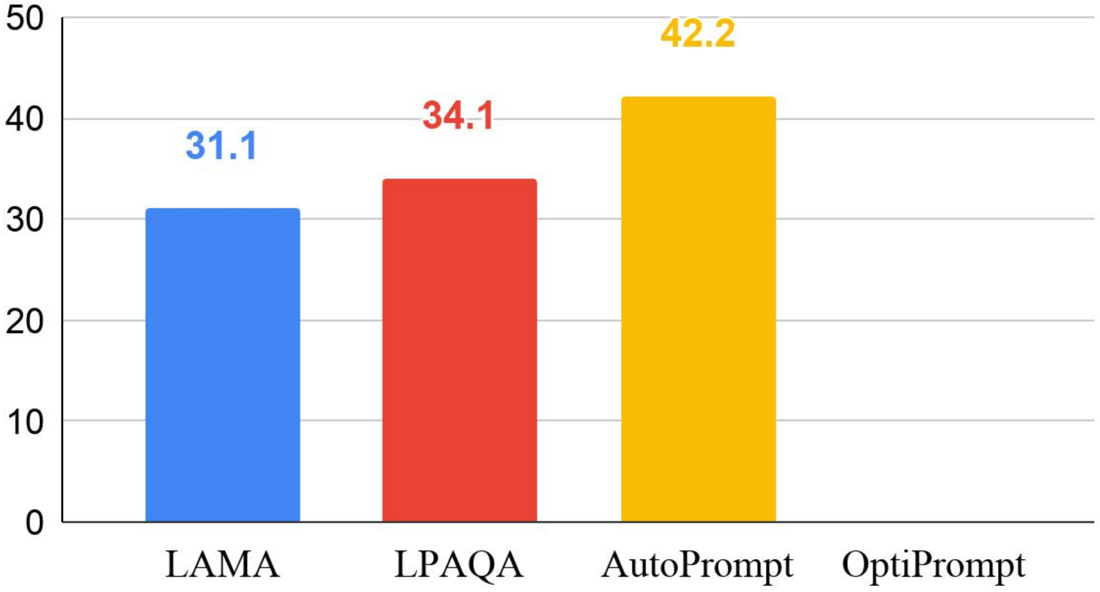
Results on the LAMA Benchmark

Results are based on BERT-base



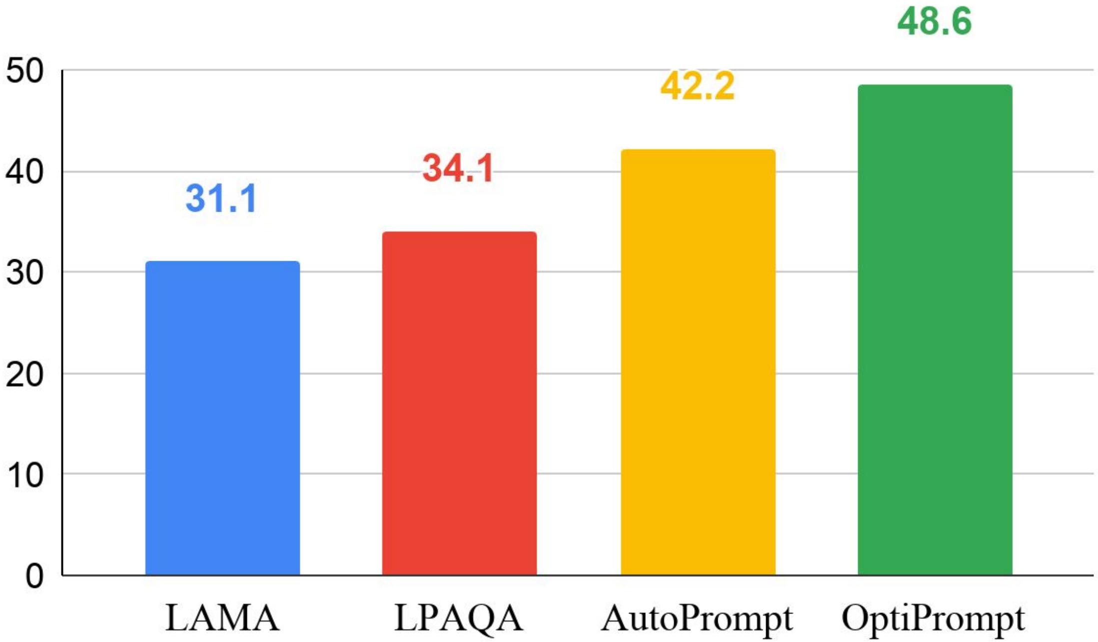
Results on the LAMA Benchmark

Results are based on BERT-base



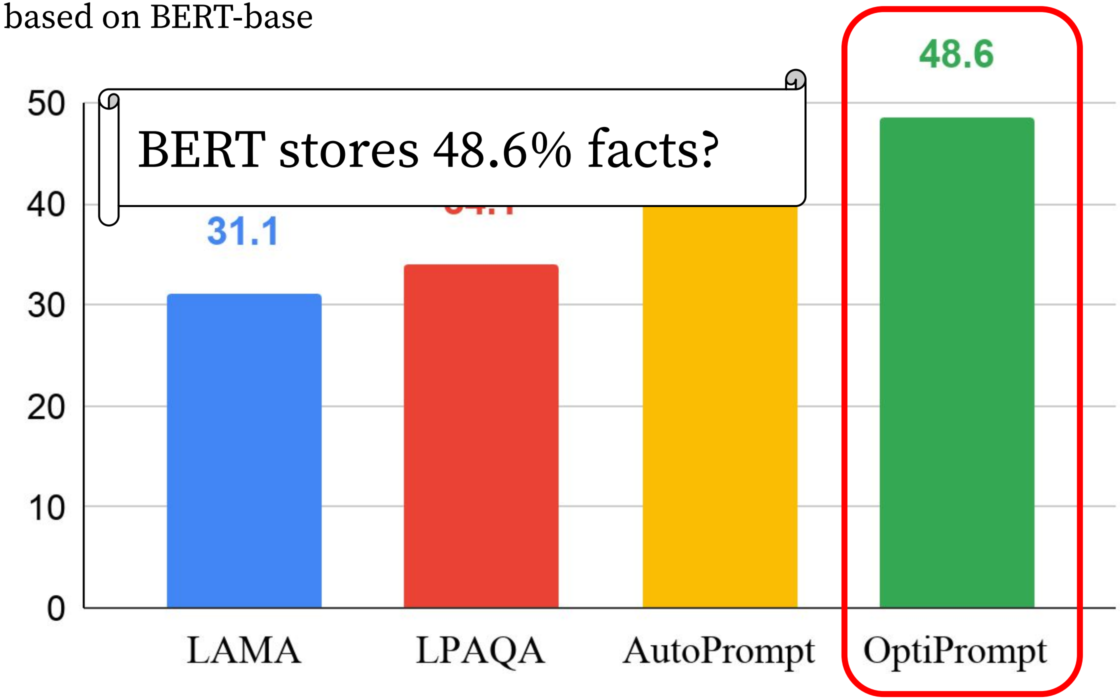
Results on the LAMA Benchmark

Results are based on BERT-base



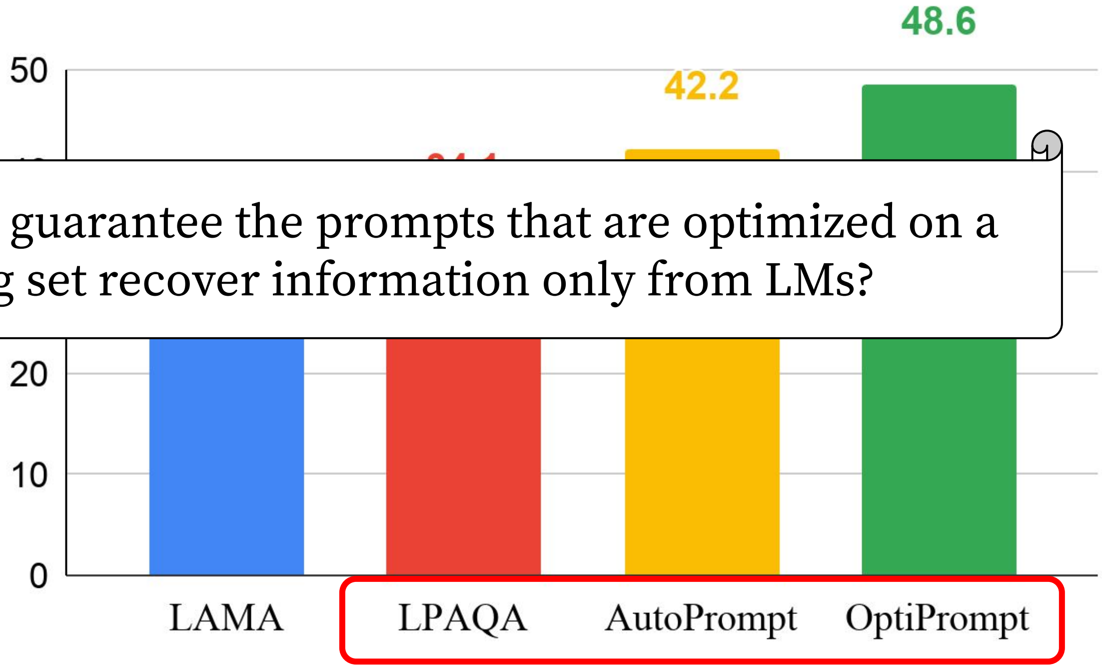
Results on the LAMA Benchmark

Results are based on BERT-base



Results on the LAMA Benchmark

Results are based on BERT-base



Can we guarantee the prompts that are optimized on a training set recover information only from LMs?

optimized on a training set

This Work

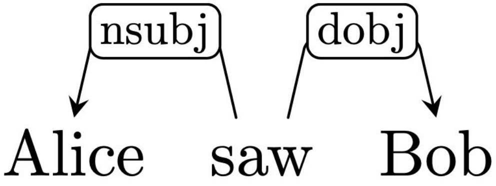
1. How to **generate** good prompts for factual probing?

2. Can we **trust** the probing results of optimized prompts?

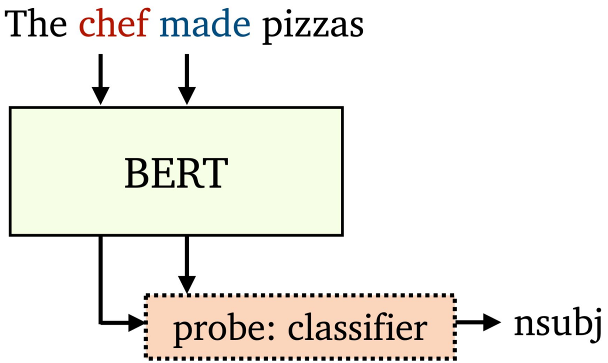
3. How can we better **interpret** the probing results?

Analogy to Linguistic Probing

Training



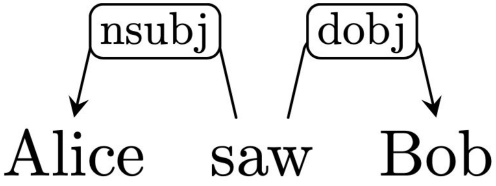
Testing



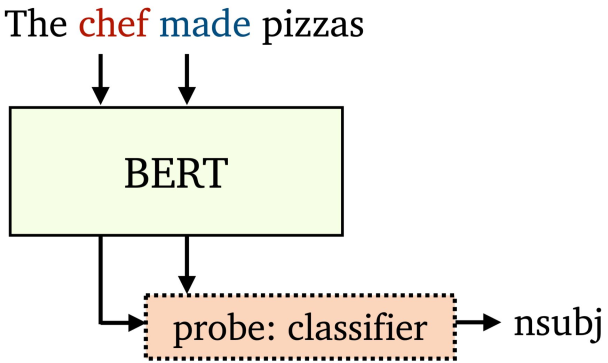
Hewitt and Liang, 2019; Pimentel et al., 2020; Voita and Titov, 2020; Zhu and Rudzicz, 2020

Analogy to Linguistic Probing

Training



Testing



Disentangle the information **encoded in the representations** from the information **learned by the probe**.

Hewitt and Liang, 2019; Pimentel et al., 2020; Voita and Titov, 2020; Zhu and Rudzicz, 2020

Training

Dante
John Donne
Guy Deghy

Florence
London
Budapest

place_of_birth

Training

Dante ⟹ Florence

John Donne ⟹ London

Guy Deghy ⟹ Budapest

place_of_birth

Testing

John Milton ⟹ ?

Are there 48.6% facts stored in the pre-trained BERT?

No!

Are there 48.6% facts stored in the pre-trained BERT?

No!

1. Unseen facts can be predicted from training data

Are there 48.6% facts stored in the pre-trained BERT?

No!

1. Unseen facts can be predicted from training data

2. Prompts can exploit training data

Facts can be predicted from training data

Majority model

- always predicts the **majority** class
- 17.3% accuracy in LAMA

Facts can be predicted from training data

Majority model

- always predicts the **majority** class
- 17.3% accuracy in LAMA

Imbalanced distributions

- native_language: **60%** French
- continent: **72%** Antarctica

Facts can be predicted from training data

Naive Bayes model

- simple **bag-of-words** classifier
- 24.6% accuracy in LAMA

Facts can be predicted from training data

Naive Bayes model

- simple **bag-of-words** classifier
- 24.6% accuracy in LAMA

Correlations between subject tokens and object tokens

- *Chevrolet* manufactures the *Chevrolet Impala*
- *Ghana Football Association* is a member of *FIFA*

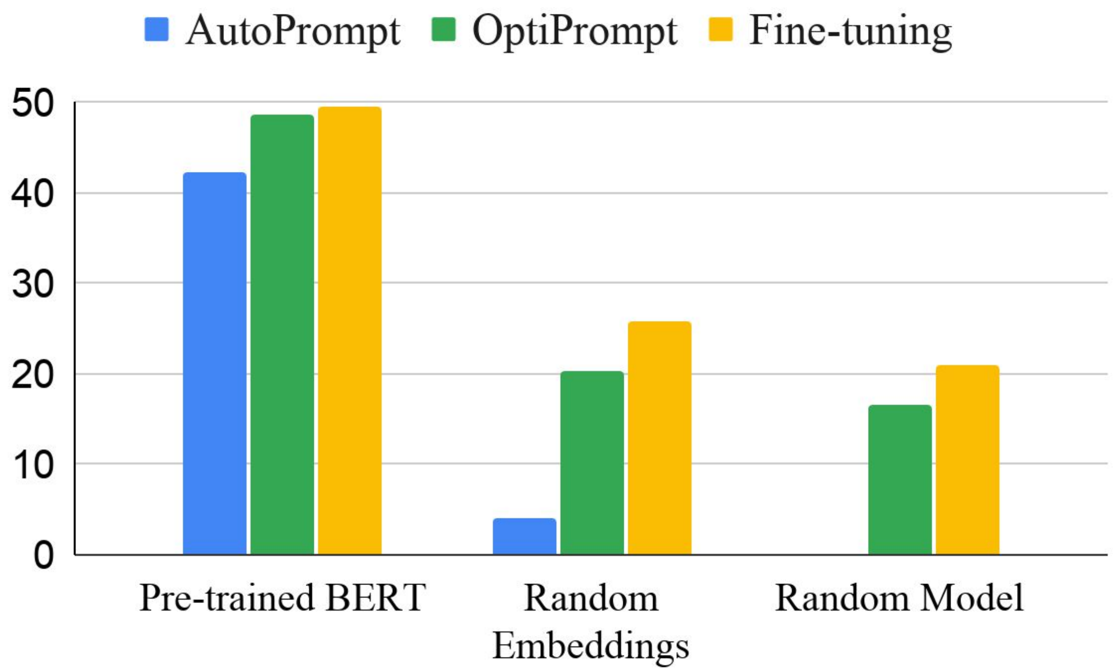
Prompts can exploit training data

Prompts can exploit training data

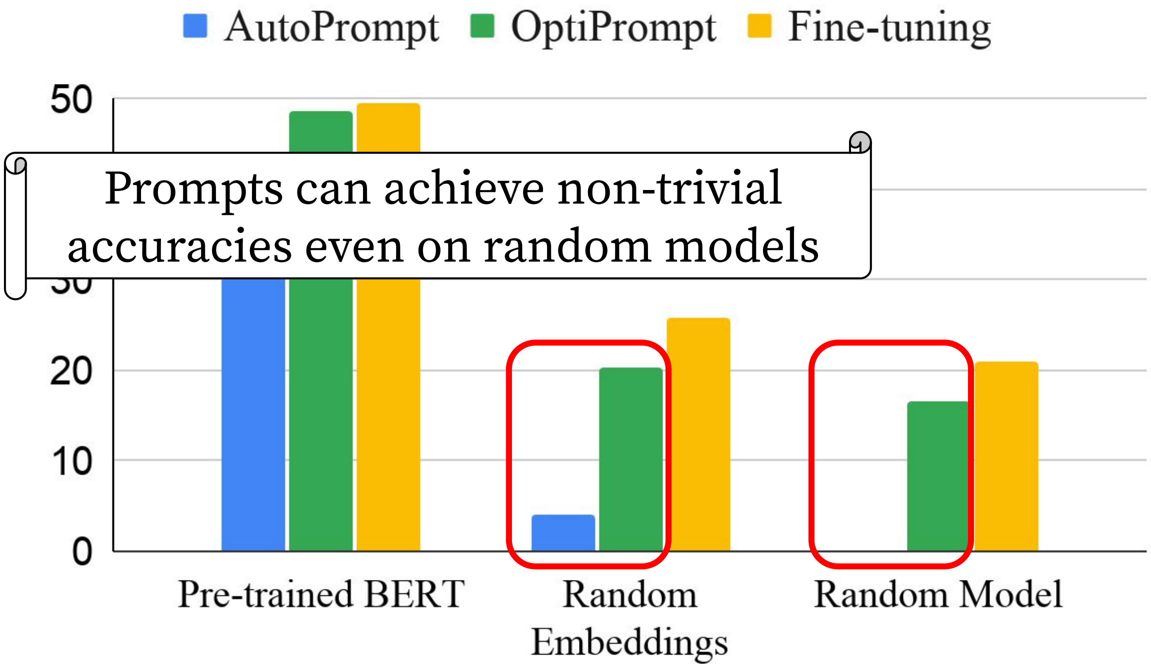
Random controls

- **Random Model**: optimize prompts on a random initialized model
- **Random Embeddings**: optimize prompts on a model with random embeddings

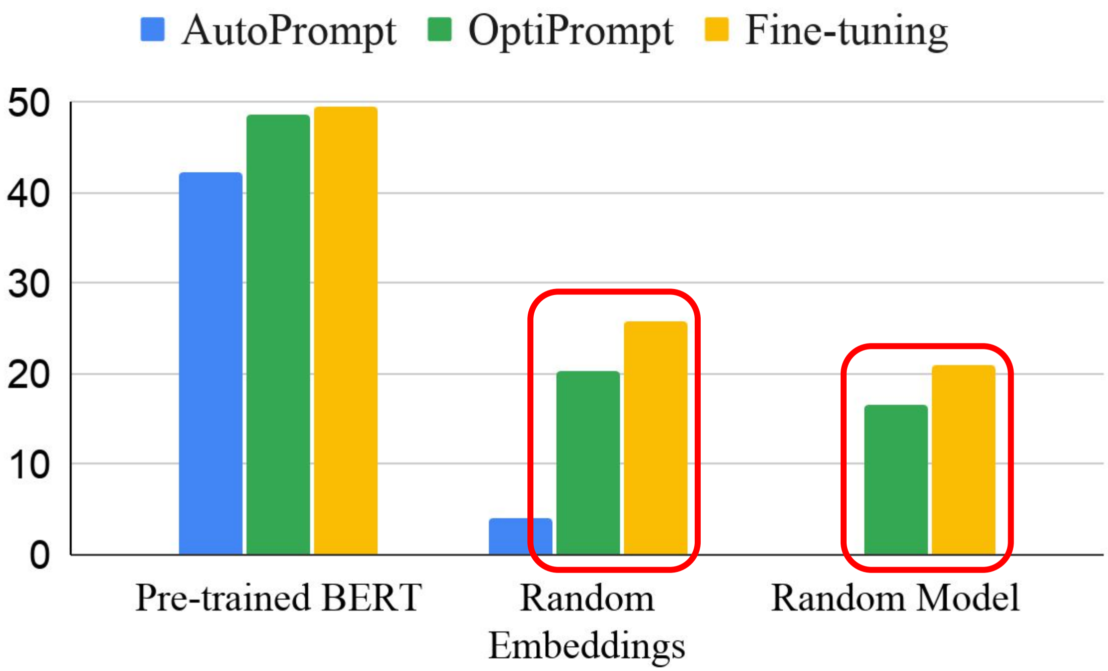
Results of random controls



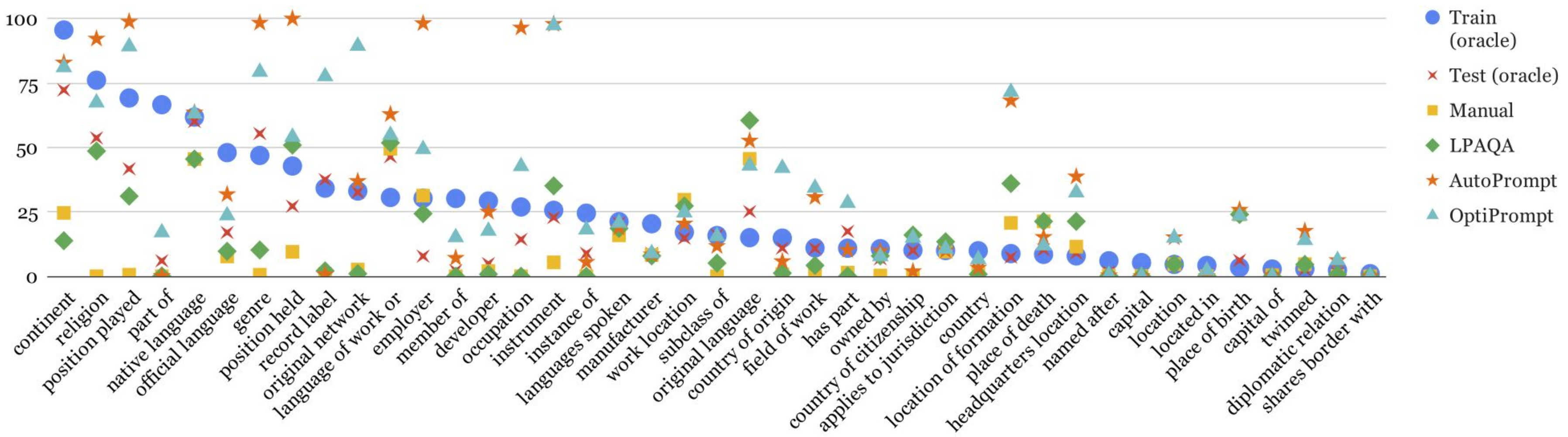
Results of random controls



Results of random controls

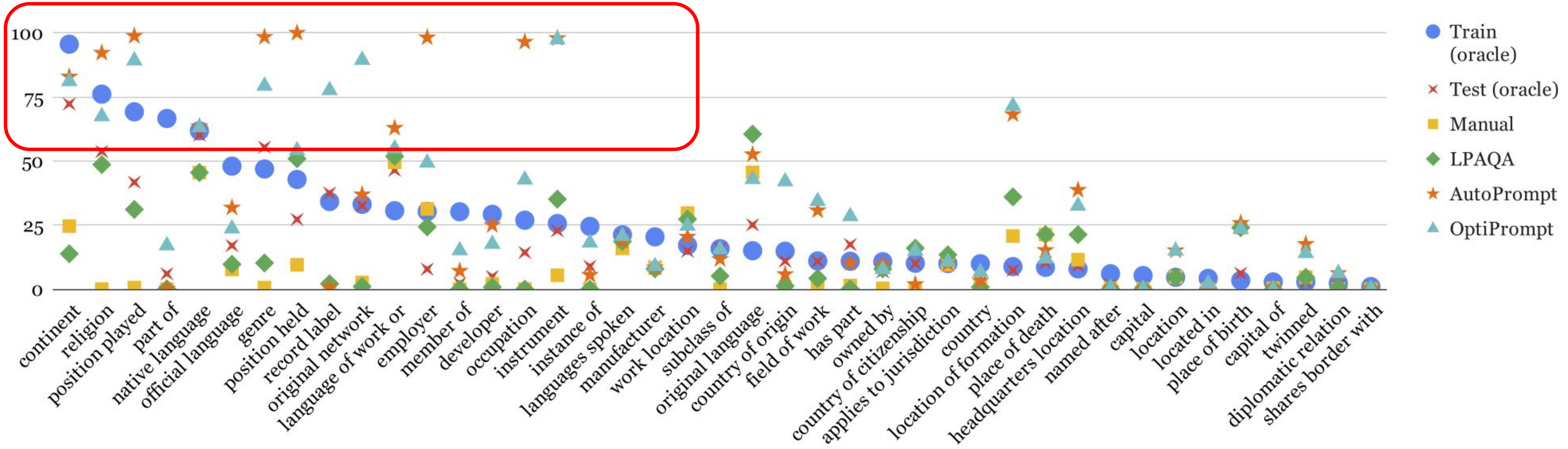


Prompts can exploit training data



Prompts can exploit training data

Over-predicting the majority class



We cannot interpret the LAMA probing results of optimized prompts as a **lower bound** of the amount of knowledge in BERT.

This Work

1. How to **generate** good prompts for factual probing?
2. Can we **trust** the probing results of optimized prompts?
3. How can we better **interpret** the probing results?

Partition LAMA examples

1. LAMA-Easy
   - Facts that can be predicted by the Naive Bayes model or by fine-tuning a random BERT on the training set

2. LAMA-Hard
   - The remain facts

Results on LAMA-Easy and LAMA-Hard

Method	**All** (34,039)	**Easy** (10,546)	**Hard** (23,493)
Manual	31.1	41.5	24.3
LPAQA	34.1	47.0	25.6
AUTOPROMPT	42.2	68.2	26.7
OPTIPROMPT	**48.6**	**75.6**	**33.0**

Results on LAMA-Easy and LAMA-Hard

Method	**All** (34,039)	**Easy** (10,546)	**Hard** (23,493)
Manual	31.1	41.5	24.3
LPAQA	34.1	47.0	25.6
AutoPrompt	42.2	68.2	26.7
OptiPrompt	**48.6**	**75.6**	**33.0**

Results on LAMA-Easy and LAMA-Hard

Method	**All** (34,039)	**Easy** (10,546)	**Hard** (23,493)
Manual	31.1	41.5	24.3
LPAQA	34.1	47.0	25.6
AUTOPROMPT	42.2	68.2	26.7
OPTIPROMPT	**48.6**	**75.6**	**33.0**

Learning from
training set

Results on LAMA-Easy and LAMA-Hard

Learning to recall facts

Method	All (34,039)	Easy (10,546)	Hard (23,493)
Manual	31.1	41.5	24.3
LPAQA	34.1	47.0	25.6
AUTOPROMPT	42.2	68.2	26.7
OPTIPROMPT	**48.6**	**75.6**	**33.0**

Learning from
training set

Conclusions

1. **OptiPrompt**: a simple & effective approach to generate prompts
2. Optimized prompts can **exploit training data** to make correct predictions
   - Probing results cannot be directly interpreted as a lower bound of amount of knowledge stored in the LM
3. **Random controls** can help us better interpret the probing results

Thank You!

Paper: https://arxiv.org/pdf/2104.05240.pdf

Code: https://github.com/princeton-nlp/OptiPrompt