Humans Learn From Task Descriptions. And So Should Our Models!

Hinrich Schütze, Timo Schick, Sahana Udupa

LMU Munich

2021-09-22

Outline

How do humans learn?

How do humans learn?

How do humans learn?

Let's look at a typical example of human learning:
How to open and eat a pomegranate

The BEST Way To Open & Eat A Pomegranate:
`https://www.youtube.com/watch?v=5BExPRwPdAs`
timestamps 10s to 45s
Read the closed captions
Pay attention to (i) descriptions, (ii) # training instances

Open & Eat A Pomegranate: What did we see?

- The teacher gives a detailed description of the task and of the solution
- Task description: way of opening/eating that is not "a pain in the butt" and not "messy"
- Solution description: "score the pomegranate along the ridges" etc.
- Very few training instances: just three

A typical form of human learning

- Detailed description
- Very few training instances (10 or fewer)

Typical machine learning setup

- No descriptions
- Large training sets
- Even few-shot learning often uses 1000s of examples

Motivation for our approach

- Humans take advantage of task descriptions.
- Machine learning models generally don't.
- How can we incorporate task descriptions into NLP?

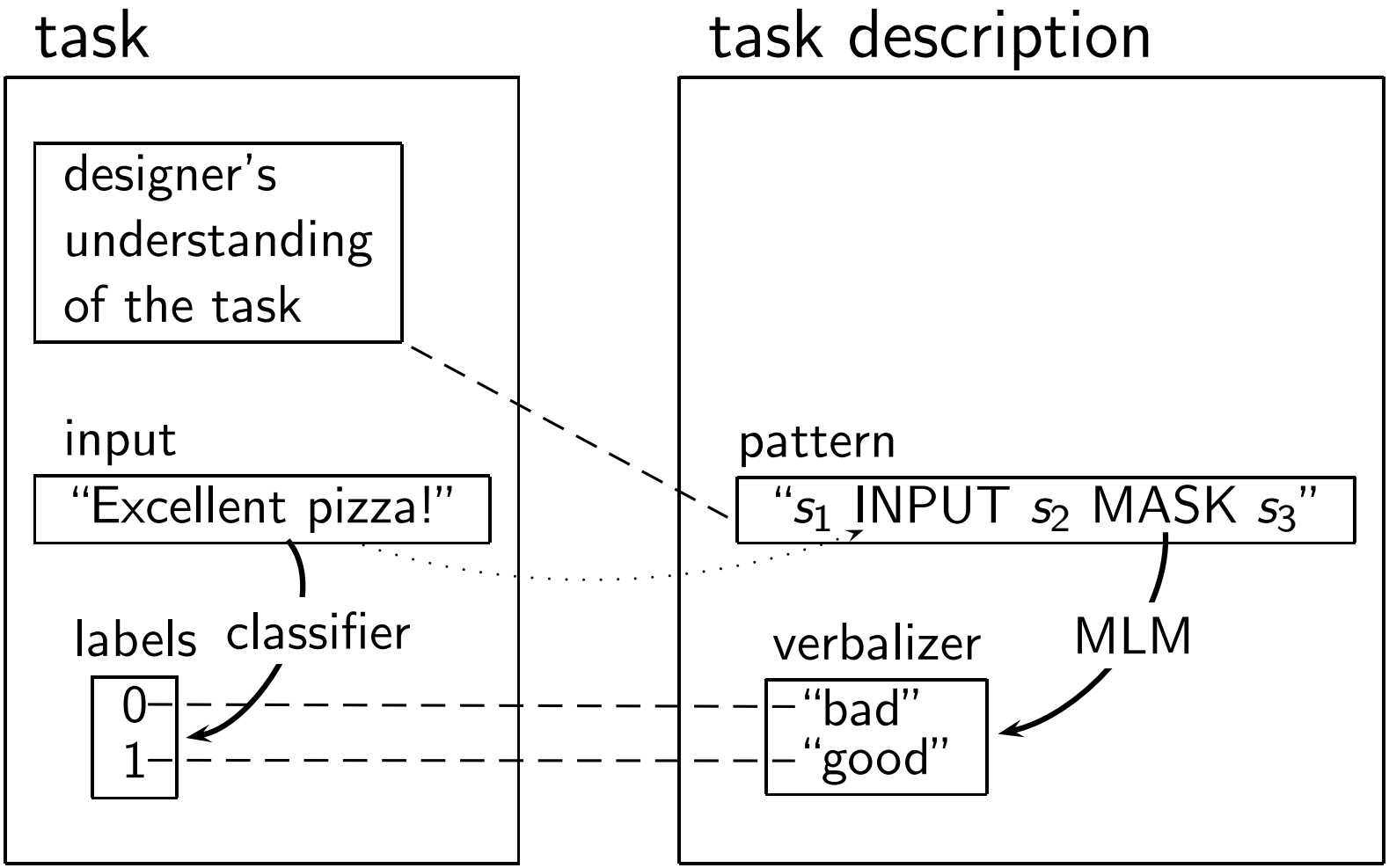
Overview

Team:

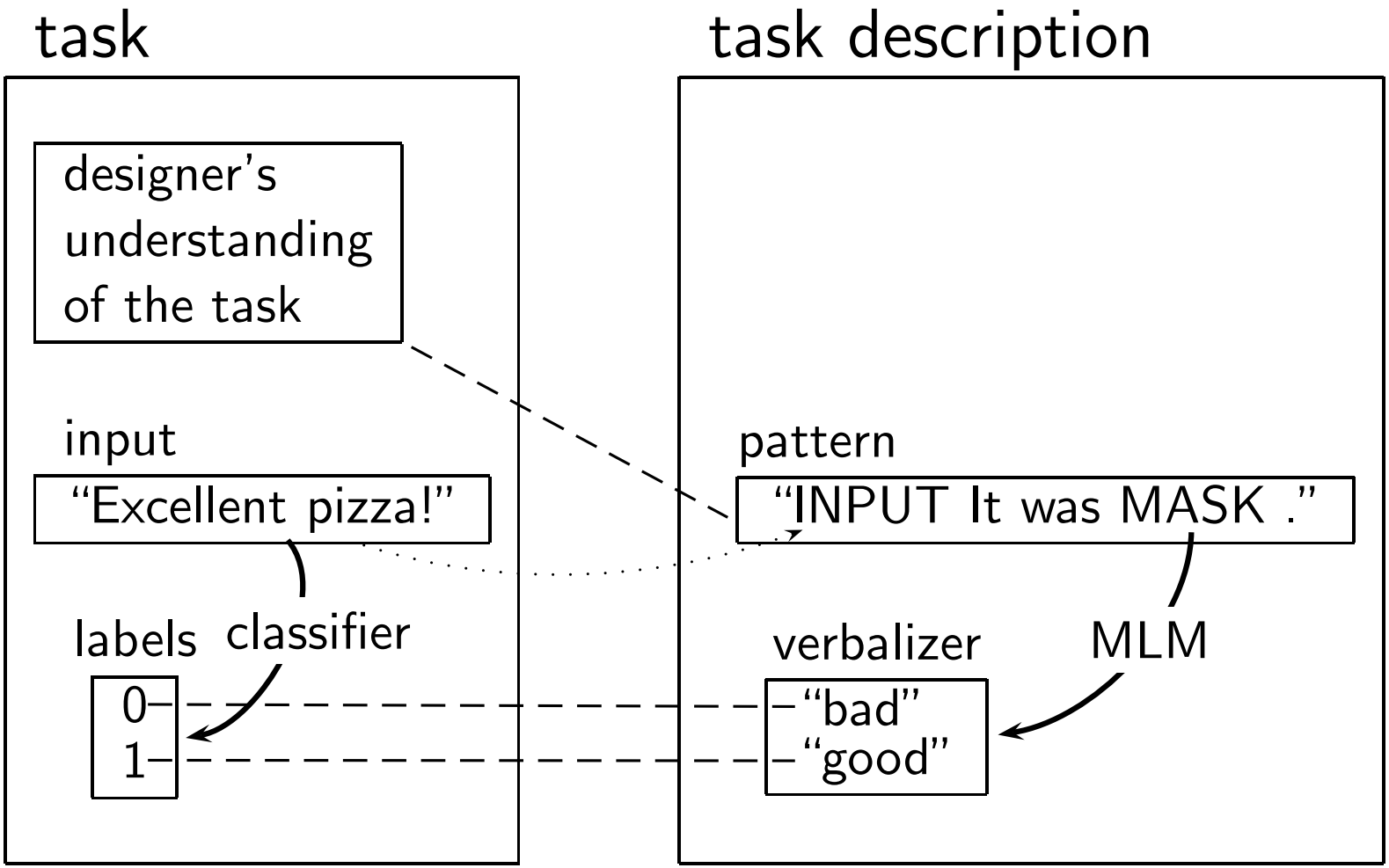
- Timo Schick (conception & actual work)
- Sahana Udupa (Professor of Media Anthropology)
- Hinrich Schütze (PhD advisor)

Outline

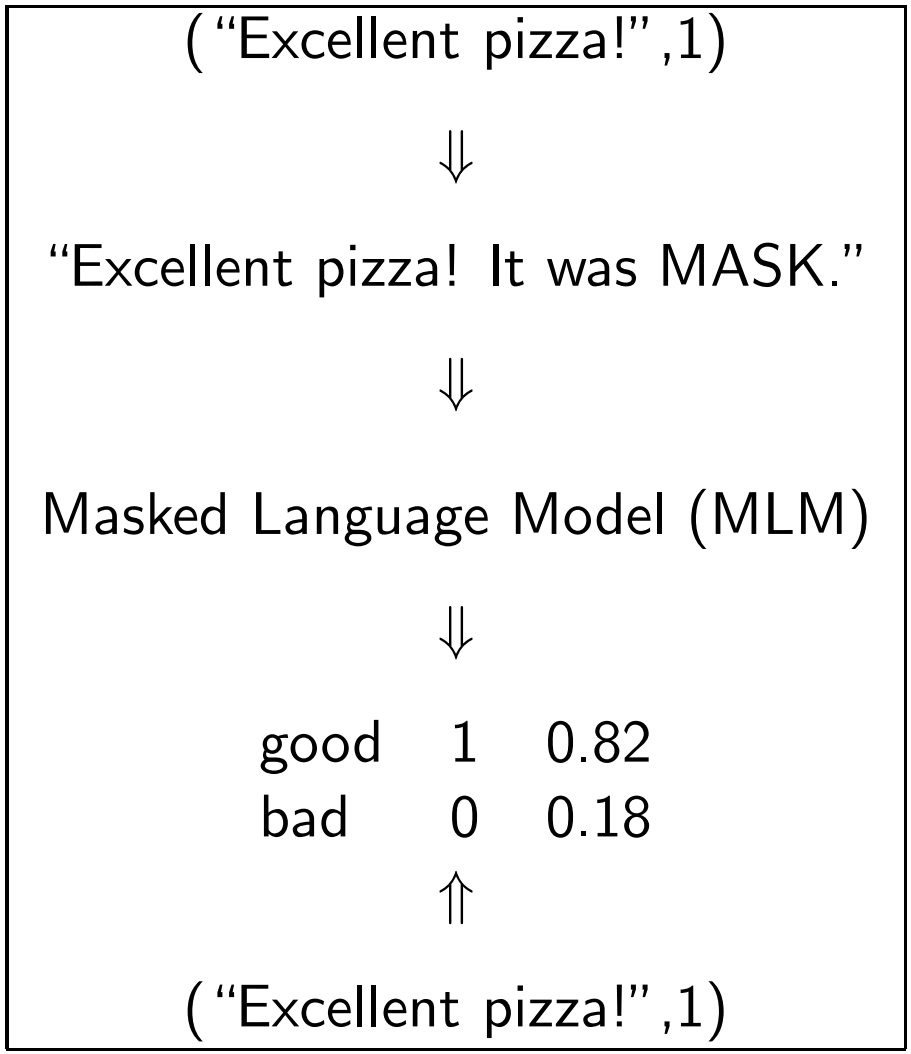
PET: Task and Task description



PET: Task and Task description



Pattern Exploiting Training (PET): Finetuning



training instance

use pattern: "*review* It was MASK."

input to MLM

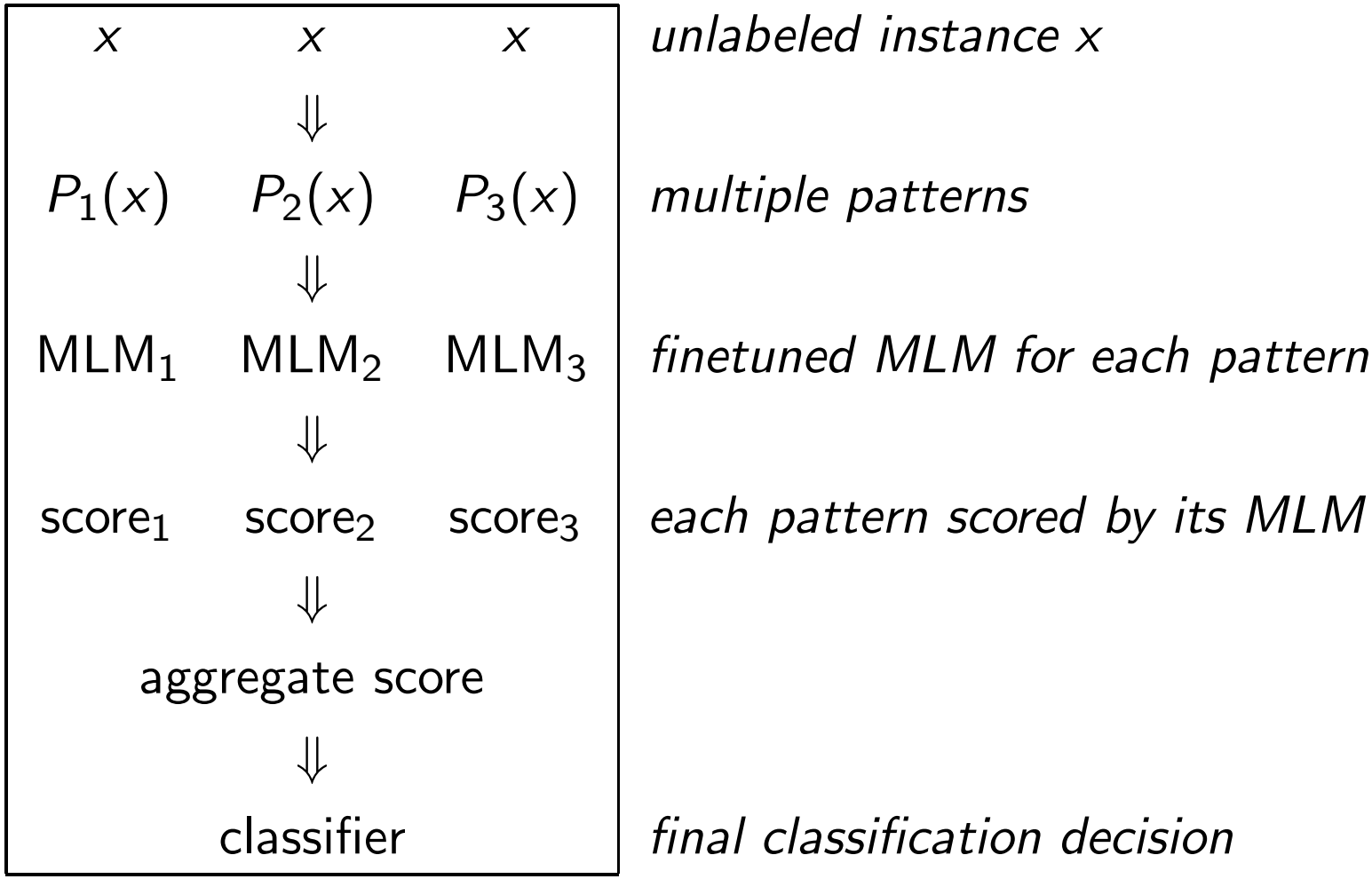
MLM predicts: $P(\begin{smallmatrix}\text{good}\\\text{bad}\end{smallmatrix}|\text{MASK})$

verbalizer

finetune MLM with cross-entropy

training instance

How to exploit multiple patterns



Multiple patterns: Example for sentiment

Verbalizer

$v(\star) =$	terrible
$v(\star\star) =$	bad
$v(\star\star\star) =$	okay
$v(\star\star\star\star) =$	good
$v(\star\star\star\star\star) =$	great

Multiple patterns: Example for sentiment

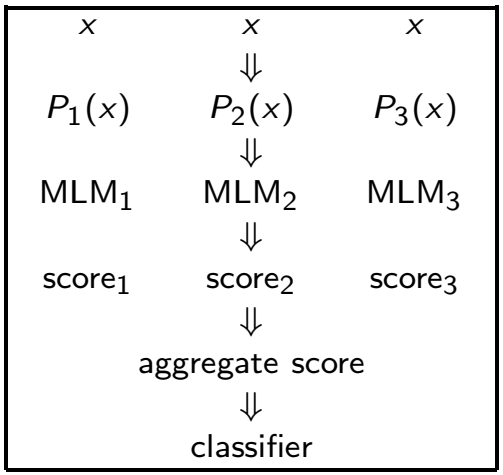
Verbalizer

$v(\star) =$	terrible
$v(\star\star) =$	bad
$v(\star\star\star) =$	okay
$v(\star\star\star\star) =$	good
$v(\star\star\star\star\star) =$	great

Patterns

$P_1(review) =$ "It was MASK. *review* "
$P_2(review) =$ "Just MASK. *review* "
$P_3(review) =$ "*review*. All in all, it was MASK."
$P_4(review) =$ "*review*. In summary, the restaurant is MASK."

Why multiple patterns are critical



unlabeled instance x

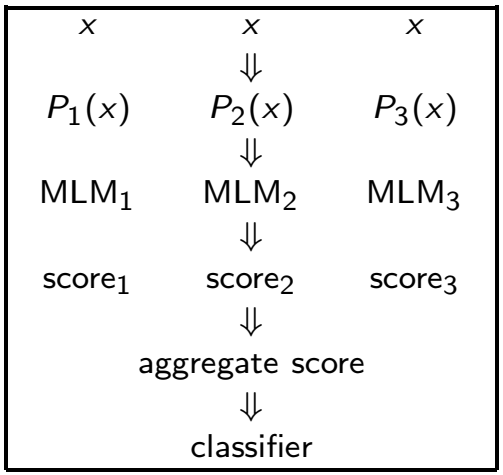
multiple patterns

finetuned MLM for each pattern

each pattern scored by its MLM

final classification decision

Why multiple patterns are critical



unlabeled instance x

multiple patterns

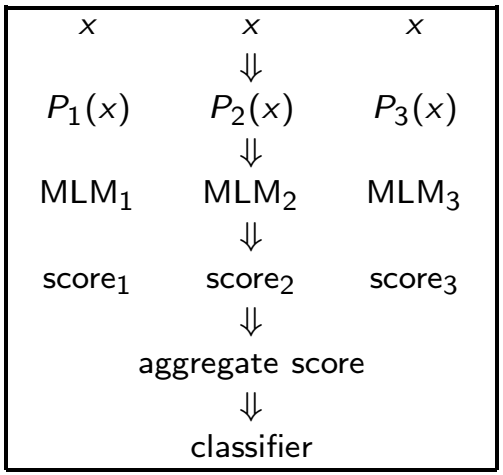
finetuned MLM for each pattern

each pattern scored by its MLM

final classification decision

- The patterns provide human expertise – the more the better!

Why multiple patterns are critical



unlabeled instance x

multiple patterns

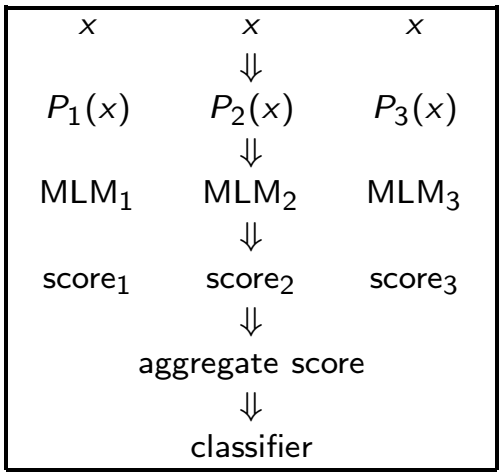
finetuned MLM for each pattern

each pattern scored by its MLM

final classification decision

- The patterns provide human expertise – the more the better!
- Realistic few-shot learning difficult without human expertise

Why multiple patterns are critical



unlabeled instance x

multiple patterns

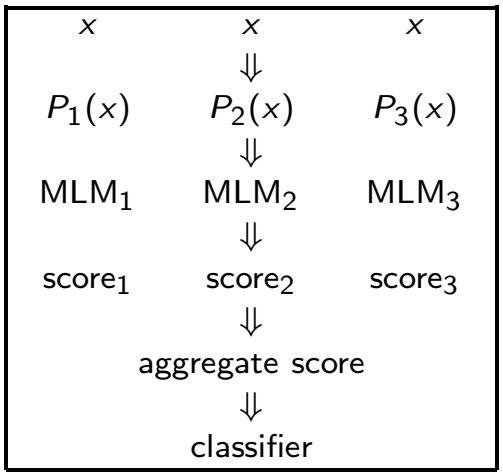
finetuned MLM for each pattern

each pattern scored by its MLM

final classification decision

- The patterns provide human expertise – the more the better!
- Realistic few-shot learning difficult without human expertise
- Can we try out multiple patterns and just keep the best one?

Why multiple patterns are critical



unlabeled instance x

multiple patterns

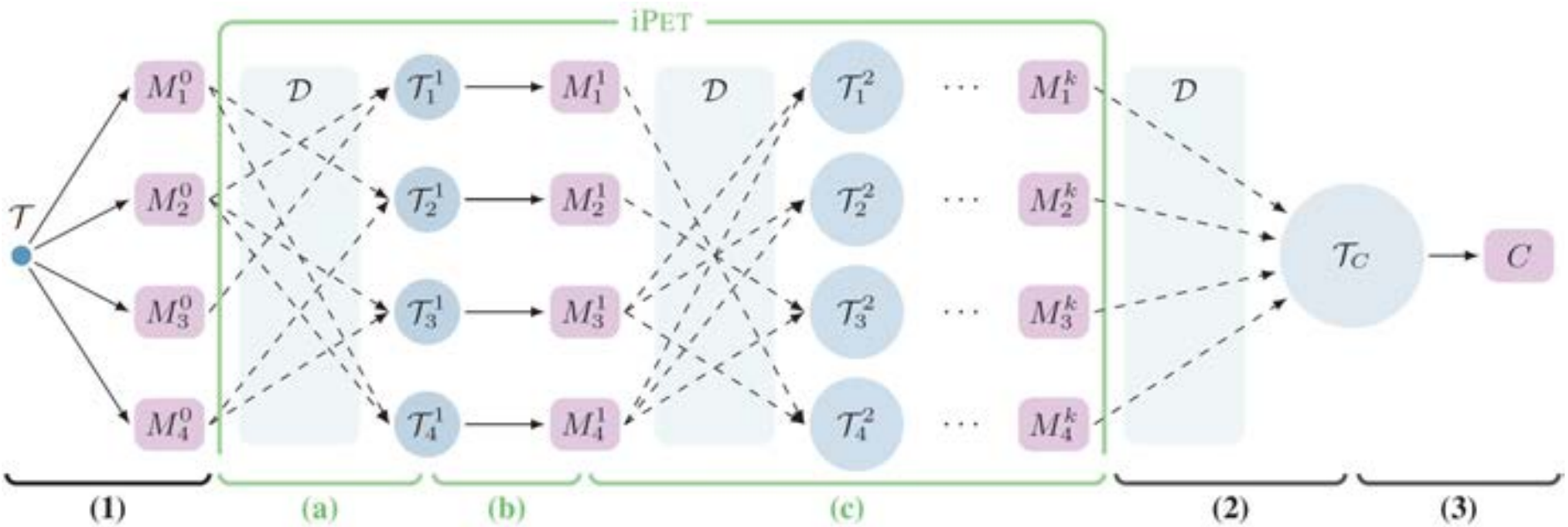
finetuned MLM for each pattern

each pattern scored by its MLM

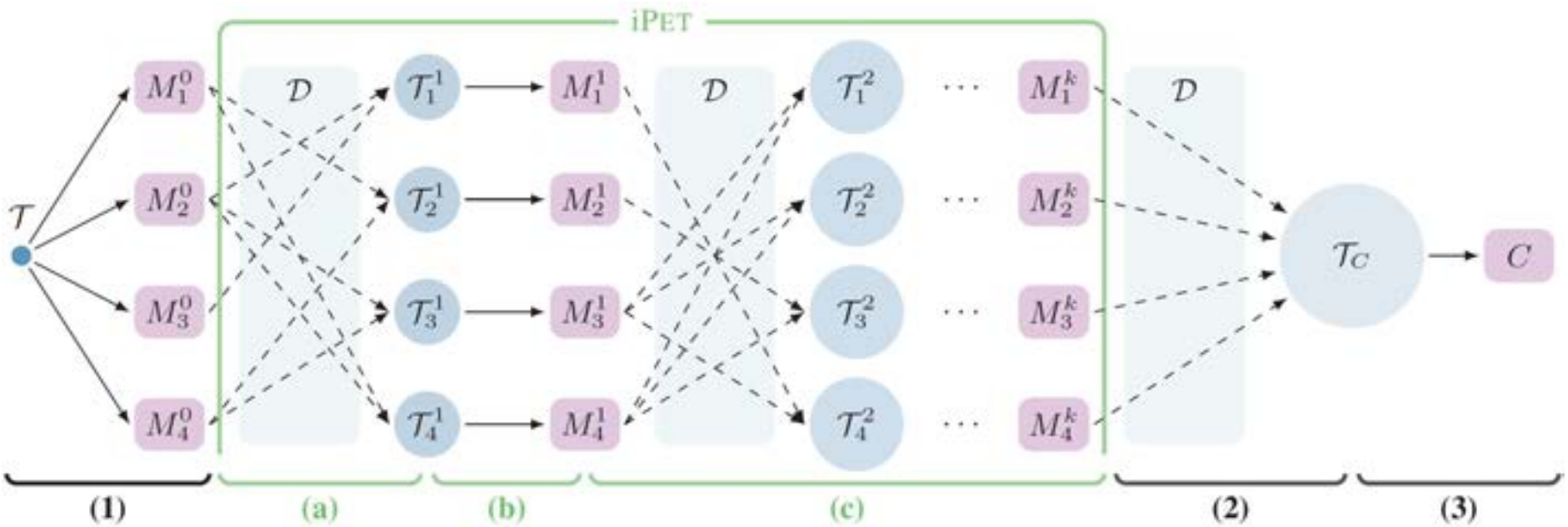
final classification decision

- The patterns provide human expertise – the more the better!
- Realistic few-shot learning difficult without human expertise
- Can we try out multiple patterns and just keep the best one?
- No: no dev set in true few-shot learning

iPET: Iterative training

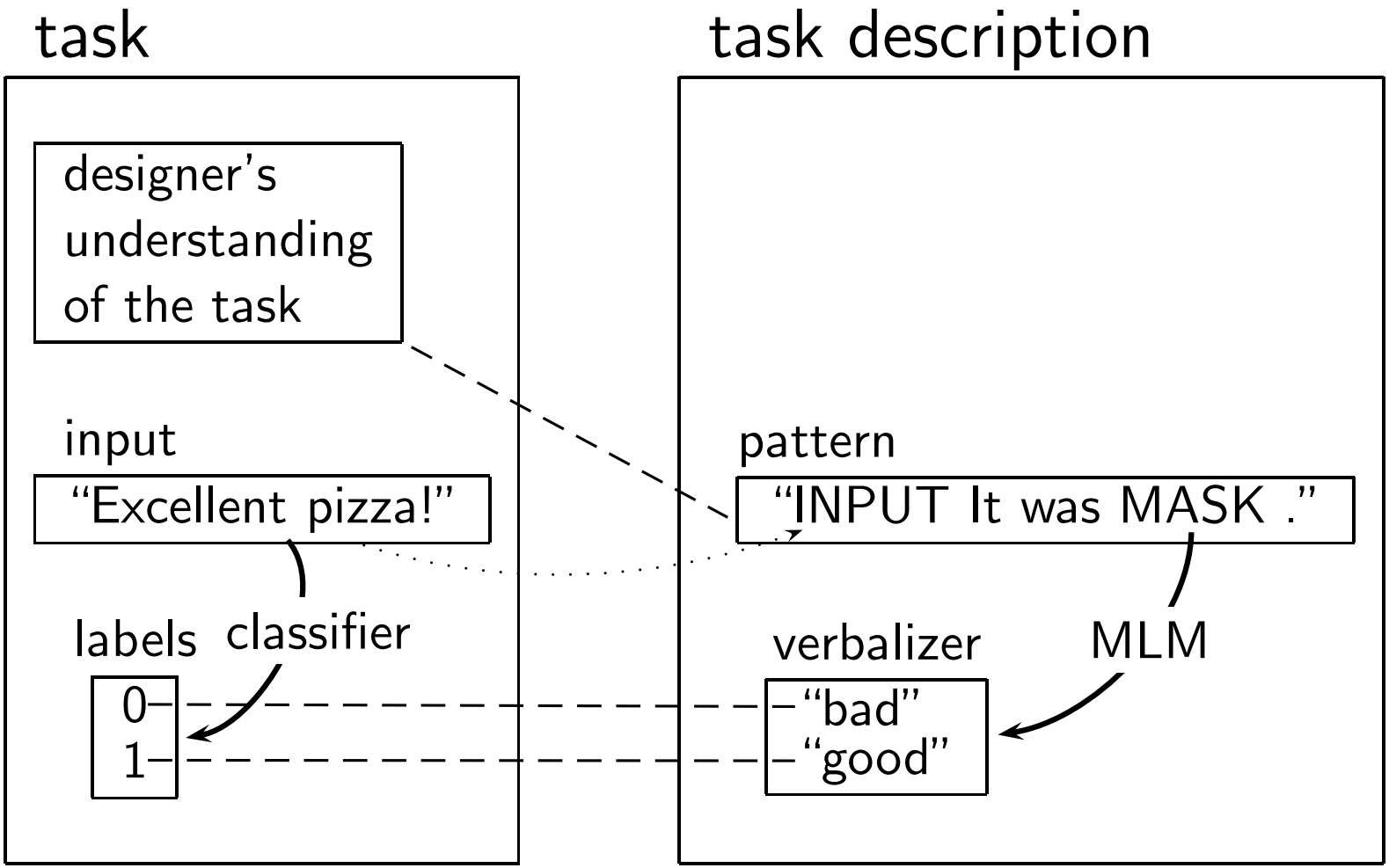


iPET: Iterative training

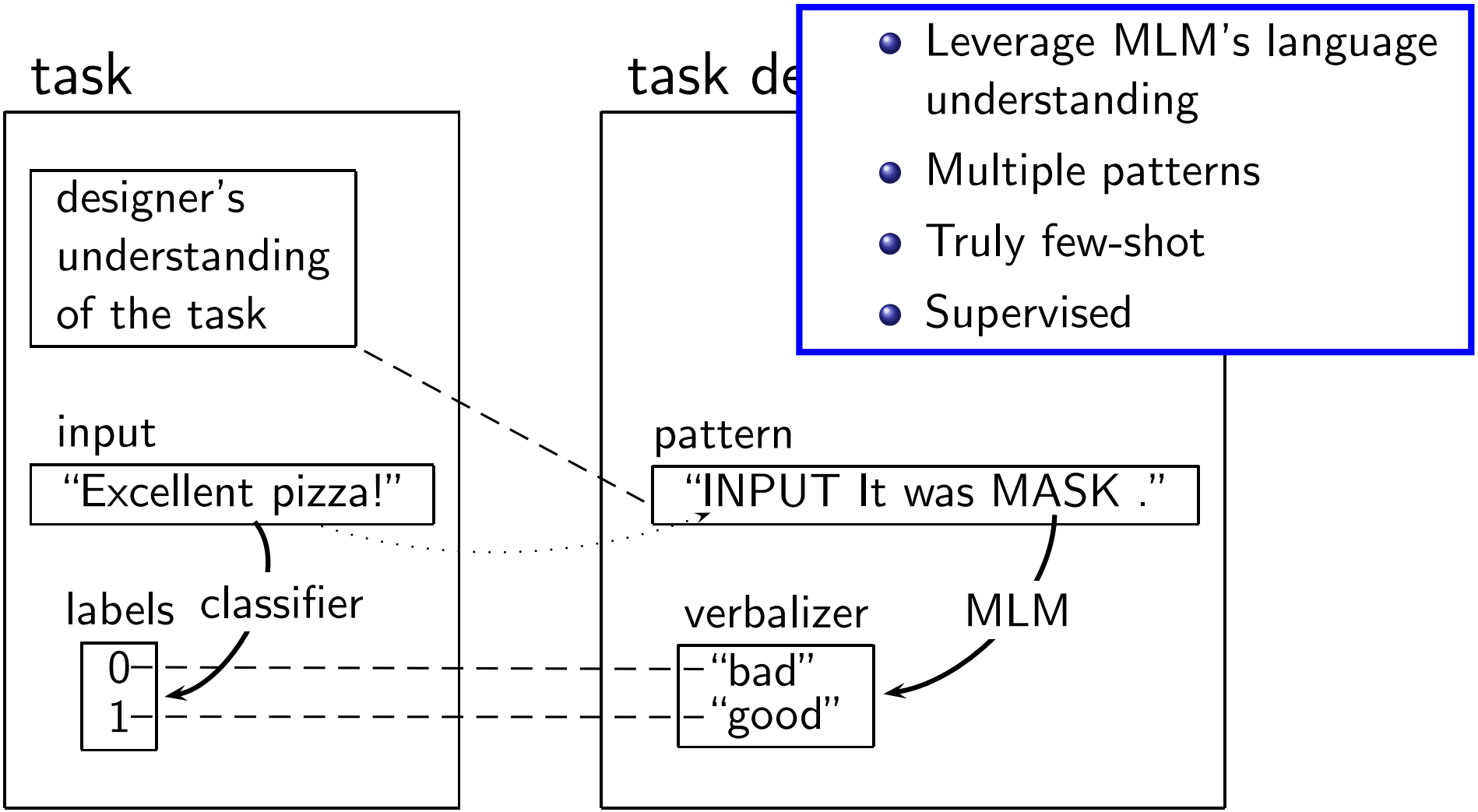


iPET = iterative PET

PET: Task and Task description



PET: Task and Task description



What exactly is a task description?

A straightforward task description

```
Translate English to French:
thanks => merci
hello => bonjour
mint => menthe
cheese =>
```

(task description)
(training instance 1)
(training instance 2)
(training instance 3)
(cloze question)

PET sentiment: Pattern and verbalizer interact

Verbalizer ("label description")

$v(\star) =$	terrible
$v(\star\star) =$	bad
$v(\star\star\star) =$	okay
$v(\star\star\star\star) =$	good
$v(\star\star\star\star\star) =$	great

Patterns

$P_1(\textit{review}) =$ "It was MASK. *review* "
$P_2(\textit{review}) =$ "Just MASK. *review* "
$P_3(\textit{review}) =$ "*review*. All in all, it was MASK."
$P_4(\textit{review}) =$ "*review*. In summary, the restaurant is MASK."

PET "Word in Context": Task description as question

Verbalizer ("label description")

$v(\text{same\_sense}) = \text{yes}$
$v(\text{different\_senses}) = \text{no}$

Pattern

$P_1(s_1, s_2, w) = s_1\ s_2$ Does w have the same meaning in both sentences? MASK

PET "Winograd Schema Challenge": No use of label descriptions

Verbalizer (not a label description)

$v(w) = w$ (identity, for all words)

Pattern

$P_1(s) = s$ In the previous sentence, the pronoun "$\star p \star$" refers to MASK.

What exactly is a task description?

What exactly is a task description?

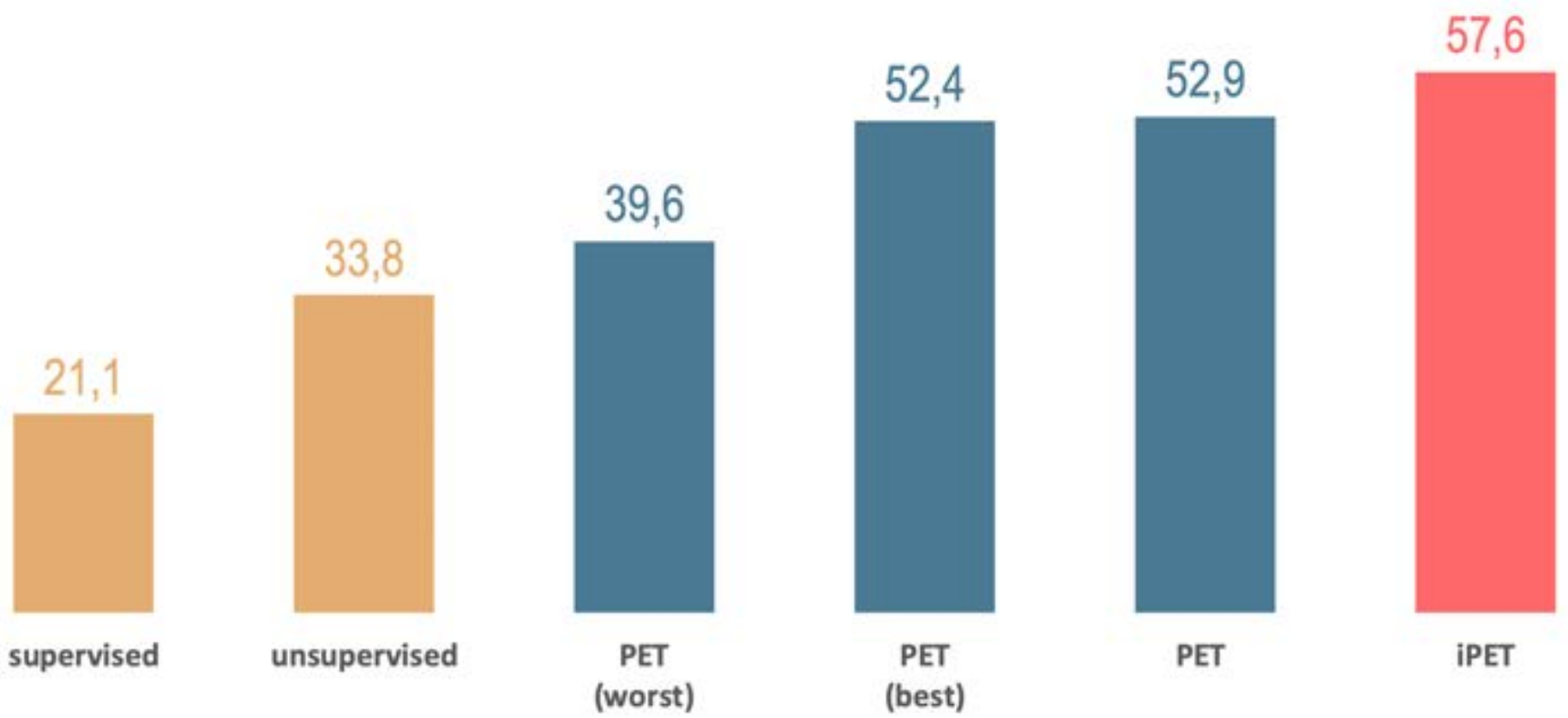
- Not a simple description of the task

What exactly is a task description?

- Not a simple description of the task
- Complex translation
  of the structure of the task into plain text

Outline

PET results on YELP FULL, $10 = 5 \cdot 2$ training examples



RoBERTa large

Multiple patterns: Example for sentiment

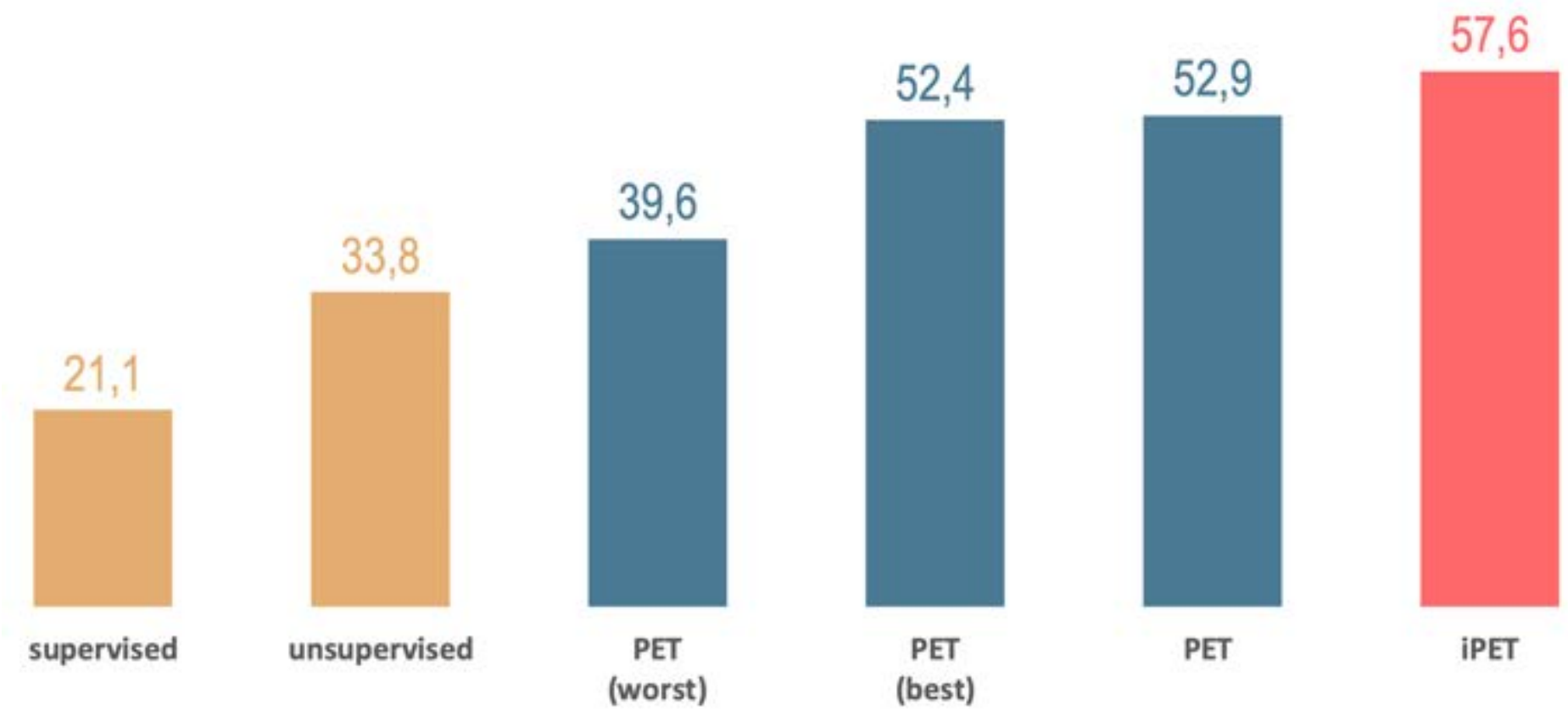
Verbalizer

$v(\star) =$ terrible
$v(\star\star) =$ bad
$v(\star\star\star) =$ okay
$v(\star\star\star\star) =$ good
$v(\star\star\star\star\star) =$ great

Patterns

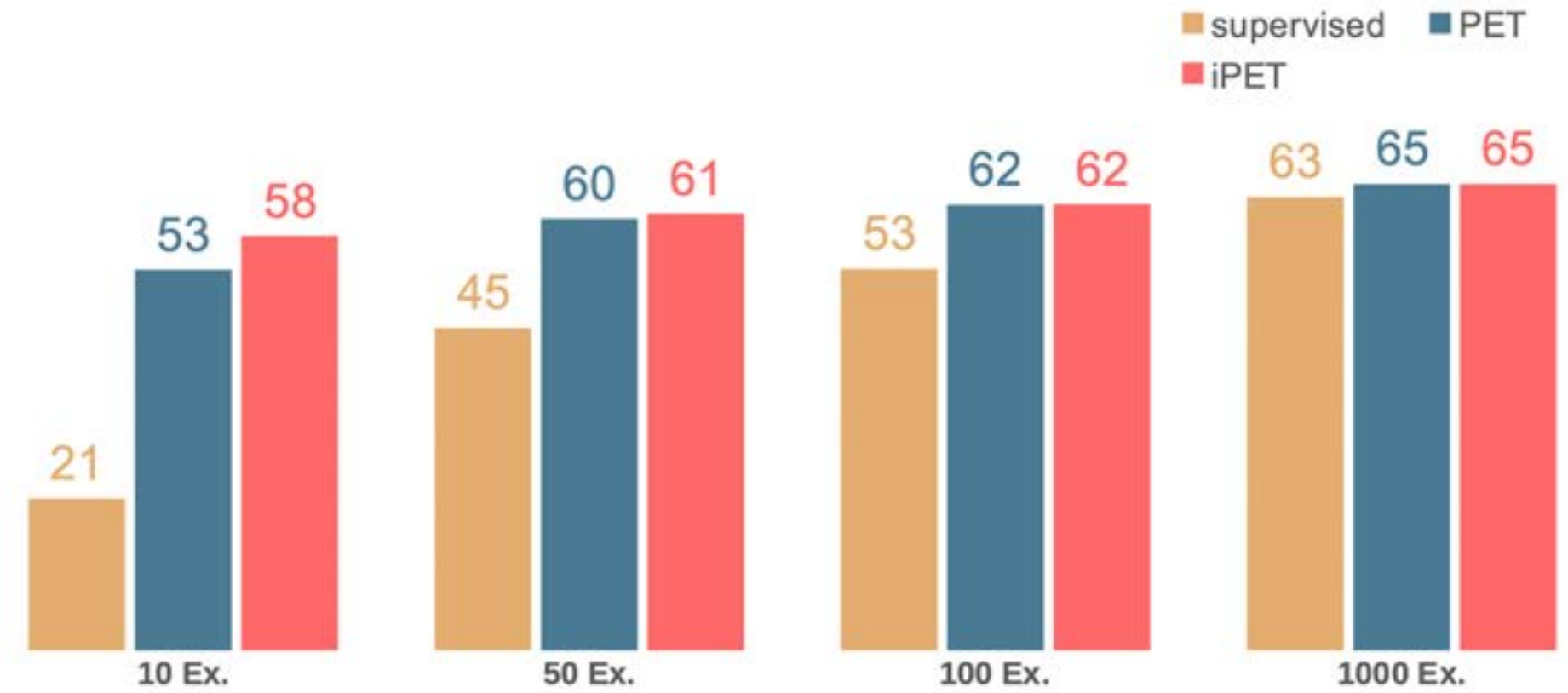
$P_1(review) =$ "It was MASK. *review* "
$P_2(review) =$ "Just MASK. *review* "
$P_3(review) =$ "*review*. All in all, it was MASK."
$P_4(review) =$ "*review*. In summary, the restaurant is MASK."

PET results on YELP FULL, $10 = 5 \cdot 2$ training examples



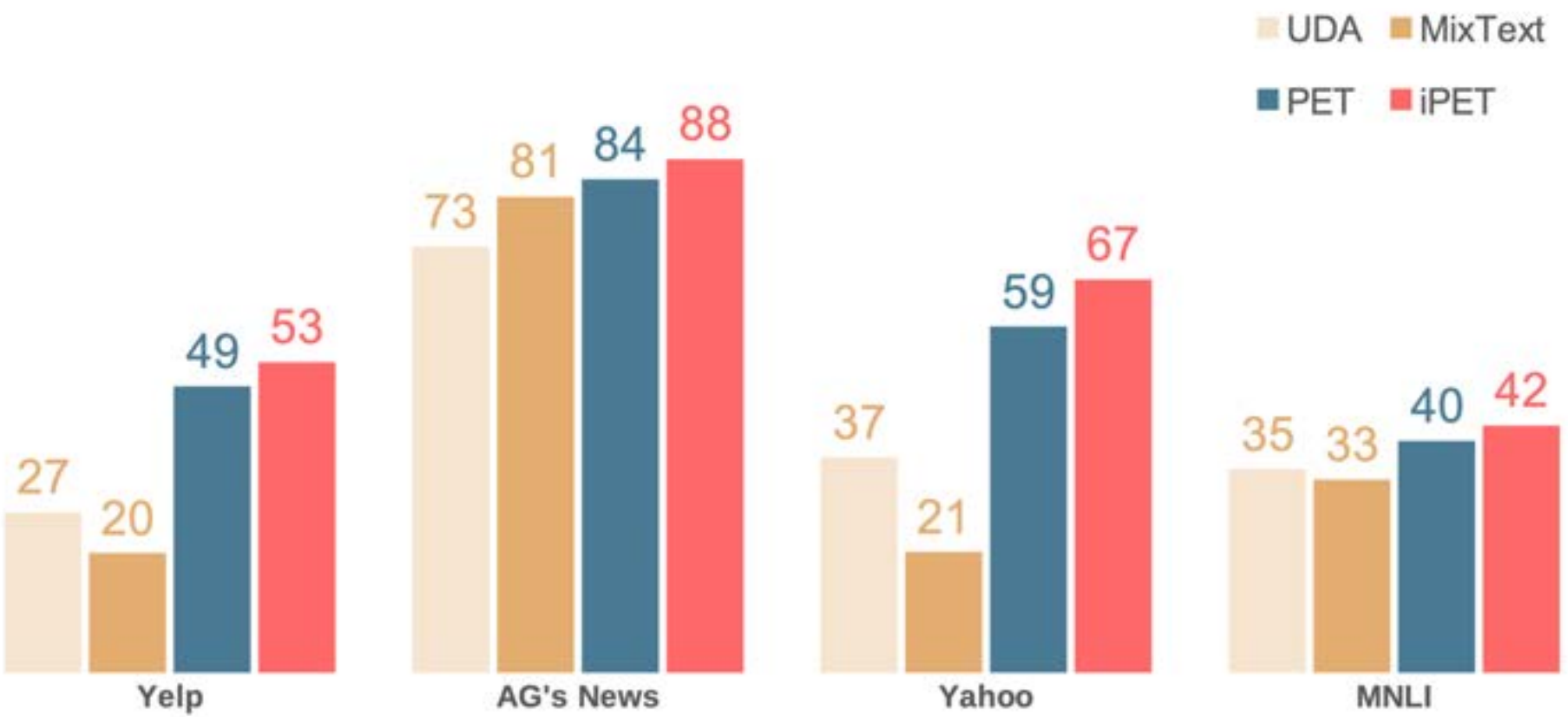
RoBERTa large

PET results on YELP FULL, effect of training set size



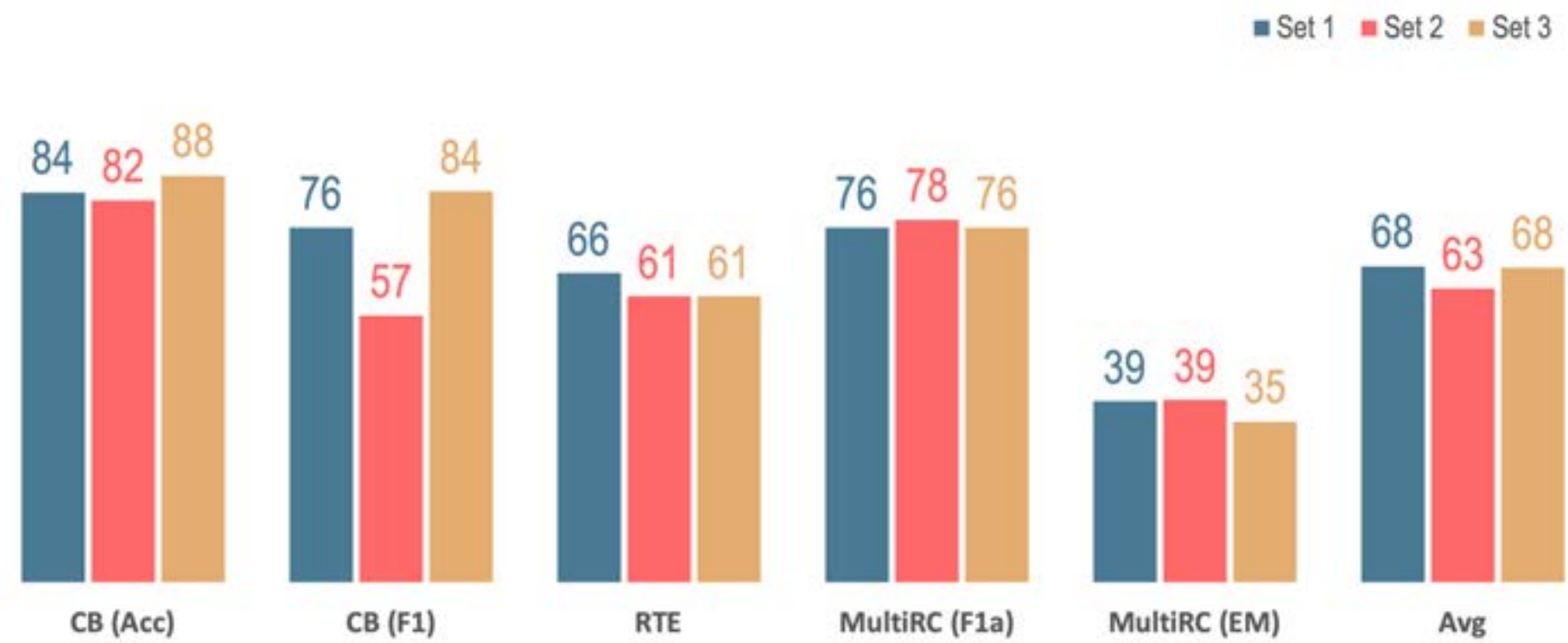
RoBERTa large

PET/iPET vs. UDA/MixText, 10 training examples



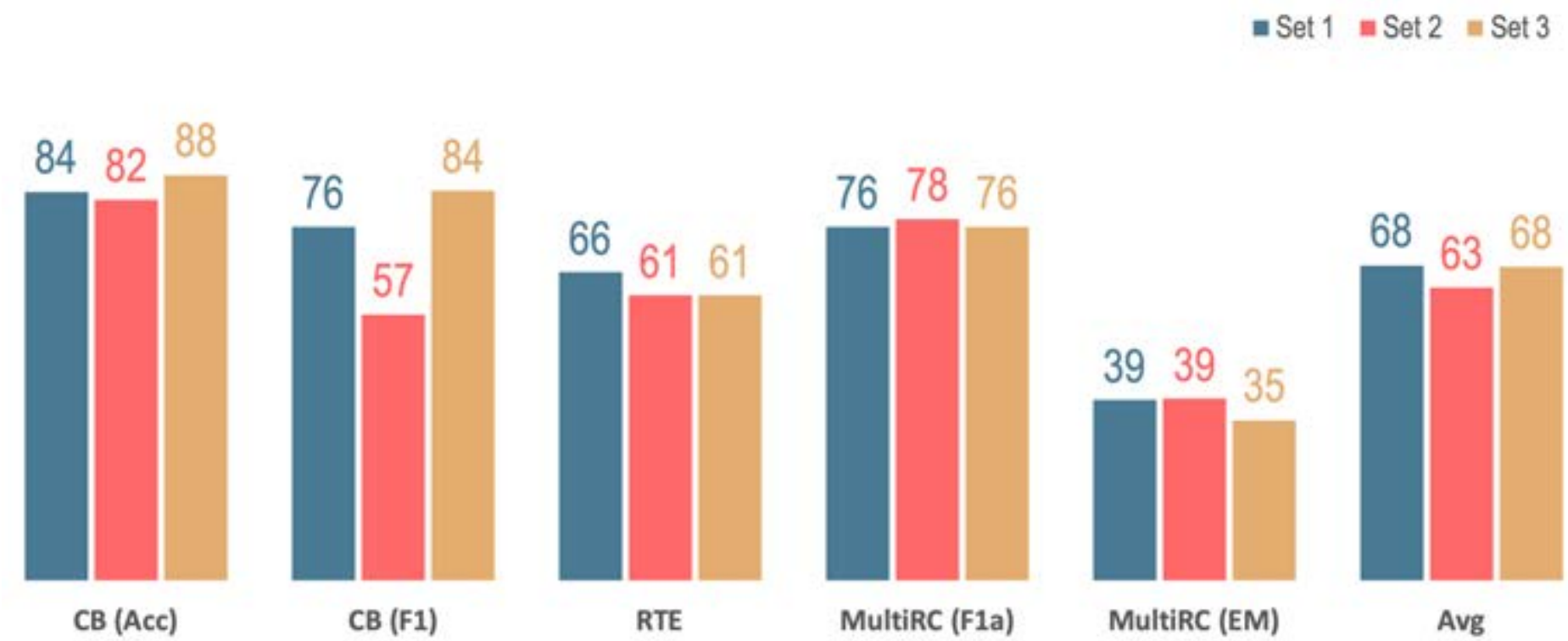
RoBERTa base

Different sets of 32 training examples: The choice of shots matters



ALBERT xxlarge

Different sets of 32 training examples: The choice of shots matters



Mengjie Zhao, Yi Zhu, Ehsan Shareghi, Roi Reichart, Anna Korhonen, Hinrich Schütze, *A Closer Look at Few-Shot Crosslingual Transfer: The Choice of Shots Matters*, ACL 2021

ALBERT xxlarge

GPT3

- Key innovation:
  No supervised finetuning for a specific task
- Instead: "in-context learning". Example:
  - Task description
  - A few training instances
  - A cloze question
- No parameter updates during in-context learning
- $\rightarrow$ No real learning takes place for a specific task.

GPT3 vs. Supervised learning

- Arguably, humans do do parameter updates when they learn.
- E.g., you don't start from scratch when you open a second pomegranate a day later.
- In contrast, GPT3 arguably doesn't learn anything after the completion of pretraining!
- So why not use: both task description and supervised learning?
- Which is what humans do . . .

GPT3 vs. Supervised learning

- Arguably, humans do do parameter updates when they learn.
- E.g., you don't start from scratch when you open a second pomegranate a day later.
- In contrast, GPT3 arguably doesn't learn anything after the completion of pretraining!
- So why not use: both task description and supervised learning?
- Which is what humans do . . .

GPT3 vs. Supervised learning

- Arguably, humans do do parameter updates when they learn.
- E.g., you don't start from scratch when you open a second pomegranate a day later.
- In contrast, GPT3 arguably doesn't learn anything after the completion of pretraining!
- So why not use: both task description and supervised learning?
- Which is what humans do . . .
- PET: 2020-01-21, GPT3: 2020-05-28

iPET vs. GPT3: Size of model

model	# params	
GPT3	175G	100.0%
GPT3 med	350M	0.2%
iPET	223M	0.1%

ALBERT xxlarge

iPET vs. GPT3: Size of model

model	# params	
GPT3	175G	100.0%
GPT3 med	350M	0.2%
iPET	223M	0.1%

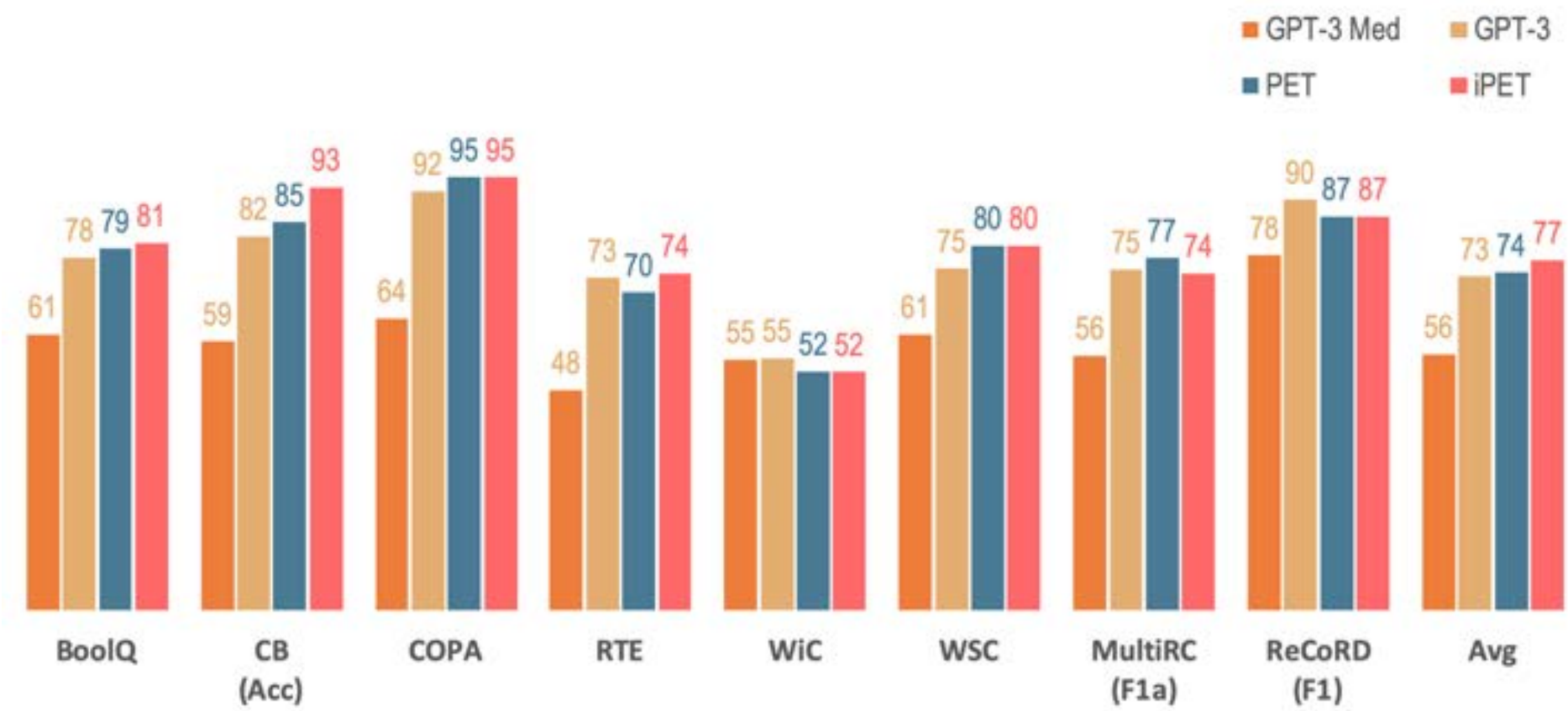
ALBERT xxlarge

iPET vs. GPT3: Size of model

model	# params	
GPT3	175G	100.0%
GPT3 med	350M	0.2%
iPET	223M	0.1%

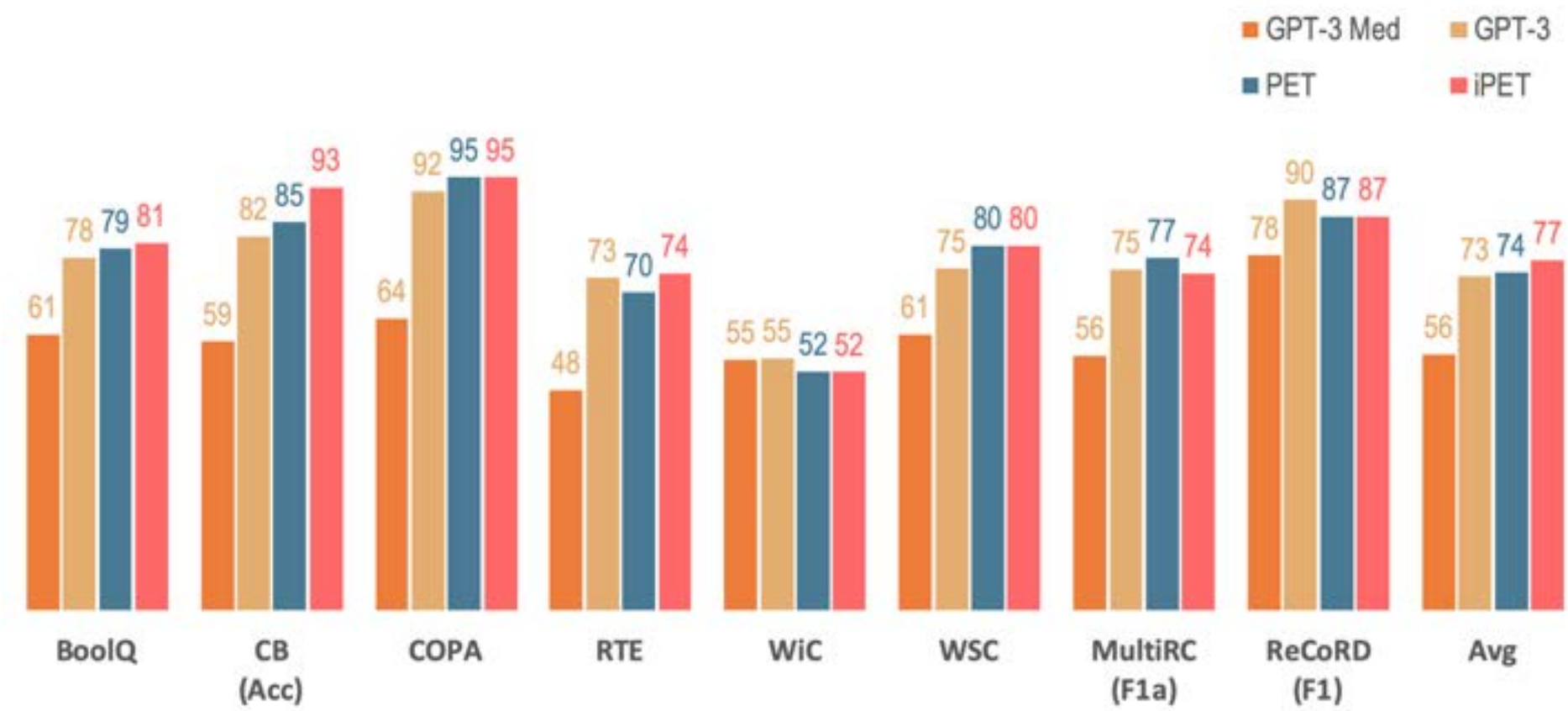
ALBERT xxlarge

PET vs. GPT3 on SuperGLUE, 32 training examples



ALBERT xxlarge

PET vs. GPT3 on SuperGLUE, 32 training examples



ALBERT xxlarge

Outline

Using patterns for debiasing

- Self-diagnosis
- Self-debiasing

Self-Diagnosis: Basic idea

Self-Diagnosis: Basic idea

- Upon self-reflection, people generally know whether something they said is considered biased.

Self-Diagnosis: Basic idea

- Upon self-reflection, people generally know whether something they said is considered biased.
- Do language models have a similar capability?

Self-Diagnosis: Experiment

pattern for self-diagnosis

"**x**"
Question: Does the above text contain **y**?
Answer:

Self-Diagnosis: Experiment

pattern for self-diagnosis

"**x**"
Question: Does the above text contain **y**?
Answer:

filled-in pattern

"I'm going to hunt you down!"
Question: Does the above text contain *a threat*?
Answer:

Self-Diagnosis: Experiment

pattern for self-diagnosis

"**x**"
Question: Does the above text contain **y**?
Answer:

filled-in pattern

"I'm going to hunt you down!"
Question: Does the above text contain *a threat*?
Answer:

Language model responds: "Yes" / "No".

Self-Diagnosis: Experiment

pattern for self-diagnosis

"**x**"
Question: Does the above text contain **y**?
Answer:

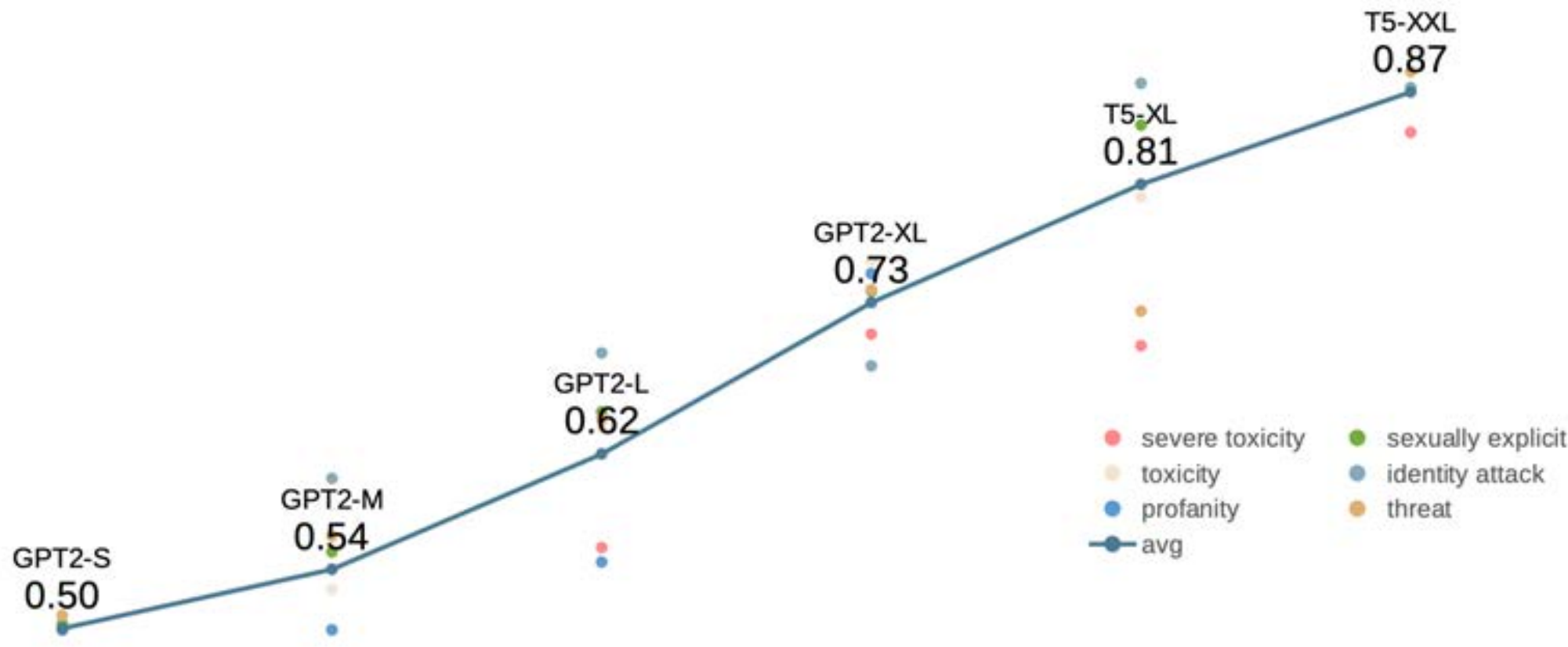
filled-in pattern

"I'm going to hunt you down!"
Question: Does the above text contain *a threat*?
Answer:

Language model responds: "Yes" / "No".

- Six emotional concepts covered by PerspectiveAPI
- RealToxicityPrompts dataset (Gehman et al, 2020)
- Top and bottom 10,000 examples for each attribute
- (Self-)Diagnosis: classify the output as biased/unbiased

Self-Diagnosis: Results



Self-Debiasing: Basic idea

- Create an evil twin:
  conditioned to produce biased output
- Compute difference between:
  - The prediction of the evil twin
  - The prediction of the unchanged LM

- Discount predictions by the difference

Self-Debiasing: Basic idea

- Create an evil twin:
  conditioned to produce biased output
- Compute difference between:
  - The prediction of the evil twin
  - The prediction of the unchanged LM

- Discount predictions by the difference

Evil twin

This slide contains offensive examples.

Evil twin

evil-twin pattern

The following text contains **y**:
x

This slide contains offensive examples.

Evil twin

evil-twin pattern

The following text contains **y**:
x

filled-in evil-twin pattern

The following text contains *gender bias*:

This slide contains offensive examples.

Evil twin

evil-twin pattern

The following text contains **y**:
x

filled-in evil-twin pattern

The following text contains *gender bias*:
Women

This slide contains offensive examples.

Evil twin

evil-twin pattern

The following text contains **y**:
x

filled-in evil-twin pattern

The following text contains *gender bias*:
Women cannot

This slide contains offensive examples.

Evil twin

evil-twin pattern

The following text contains **y**:
x

filled-in evil-twin pattern

The following text contains *gender bias*:
Women cannot code

This slide contains offensive examples.

Self-Debiasing: Basic idea

- Create an evil twin:
  conditioned to produce biased output
- Compute difference between:
  - The prediction of the evil twin
  - The prediction of the unchanged LM

- Discount predictions by the difference

Self-Debiasing: Basic idea

- Create an evil twin:
  conditioned to produce biased output
- Compute difference between:
  - The prediction of the evil twin
  - The prediction of the unchanged LM

This slide contains offensive examples.

- Discount predictions by the difference

Self-Debiasing: Basic idea

- Create an evil twin:
  conditioned to produce biased output
- Compute difference between:
  - The prediction of the evil twin
  - The prediction of the unchanged LM
  - $\Delta(w, \mathbf{x}, \mathbf{y}) =$
    $p_M(w \mid \mathbf{x}) - p_M(w \mid \text{evil-twin}(\mathbf{x}, \mathbf{y}))$
- Discount predictions by the difference

This slide contains offensive examples.

Self-Debiasing: Basic idea

- Create an evil twin:
  conditioned to produce biased output
- Compute difference between:
  - The prediction of the evil twin
  - The prediction of the unchanged LM
  - $\Delta(w, \mathbf{x}, \mathbf{y}) =$
    $p_M(w \mid \mathbf{x}) - p_M(w \mid \text{evil-twin}(\mathbf{x}, \mathbf{y}))$
- Discount predictions by the difference
- $p_M^{\text{SD}}(w \mid \mathbf{x}) \propto \alpha(\Delta(w, \mathbf{x}, \mathbf{y})) \cdot p_M(w \mid \mathbf{x})$

This slide contains offensive examples.

Self-Debiasing: Basic idea

- Create an evil twin:
  conditioned to produce biased output
- Compute difference between:
  - The prediction of the evil twin
  - The prediction of the unchanged LM
  - $\Delta(w, \mathbf{x}, \mathbf{y}) =$
    $p_M(w \mid \mathbf{x}) - p_M(w \mid \text{evil-twin}(\mathbf{x}, \mathbf{y}))$
- Discount predictions by the difference
- $p_M^{\text{SD}}(w \mid \mathbf{x}) \propto \alpha(\Delta(w, \mathbf{x}, \mathbf{y})) \cdot p_M(w \mid \mathbf{x})$
- For p_M^{SD}:
  "Women cannot code" less likely,
  "Women cannot be ordained" more likely

This slide contains offensive examples.

Results on Crows-Pairs (Nangia et al., 2020)

	BERT-base		BERT-large		RoBERTa	
Bias Type	−SD	+SD	−SD	+SD	−SD	+SD
Race / Color	58.1	54.5 ↓	60.1	54.1 ↓	64.2	52.3 ↓
Gender	58.0	51.9 ↓	55.3	54.2 ↓	58.4	54.2 ↓
Occupation	59.9	60.5 ↑	56.4	51.2 ↓	66.9	64.5 ↓
Nationality	62.9	53.5 ↓	52.2	50.1 ↓	66.7	66.0 ↓
Religion	71.4	66.7 ↓	68.6	66.7 ↓	74.3	67.7 ↓
Age	55.2	48.3 ↓	55.2	57.5 ↑	71.3	64.4 ↓
Sexual orient.	67.9	77.4 ↑	65.5	69.1 ↑	64.3	67.9 ↑
Physical app.	63.5	52.4 ↓	69.8	61.9 ↓	73.0	58.7 ↓
Disability	61.7	66.7 ↑	76.7	75.0 ↓	70.0	63.3 ↓
Average	60.5	56.8 ↓	59.7	56.4 ↓	65.5	58.8 ↓

Self-Diagnosis and Self-Debiasing

- LMs have a limited understanding of their own biases.
- We can exploit this for self-diagnosis and self-debiasing.
- Does not solve the problem of bias in NLP models,
  but could be one tool in our toolbox.

Outline

Related work

[Ben-David et al., 2021] [Betz and Richardson, 2020] [Betz et al., 2021]
[Bowman and Dahl, 2021] [Chen et al., 2020] [Chen et al., 2021]
[Dong et al., 2021] [Du et al., 2021] [Duval et al., 2021]
[Gao et al., 2020] [Guo et al., 2020] [Hambardzumyan et al., 2021]
[Han et al., 2021] [Haviv et al., 2021] [Hedderich et al., 2021]
[Herzig et al., 2021] [Holtzman et al., 2021] [Ju et al., 2021]
[Kaneko and Bollegala, 2021] [Lanchantin et al., 2020] [Lee et al., 2021]
[Lester et al., 2021] [Li and Caragea, 2021] [Li and Liang, 2021]
[Lin et al., 2021] [Liu et al., 2021] [Lu et al., 2020] [Lu et al., 2021]
[Mishra et al., 2021] [Monaco et al., 2021] [Murty et al., 2021]
[Paolini et al., 2021] [Perez et al., 2021] [Qin and Eisner, 2021]
[Raman et al., 2020] [Rethmeier and Augenstein, 2020]
[Sabando et al., 2021] [Sarti, 2020] [Saunshi et al., 2020]
[Scao and Rush, 2021] [Shin et al., 2020] [Shin et al., 2021]
[Shorten et al., 2021] [Tam et al., 2021] [Tang et al., 2021]
[Wang and Scott, 2021] [Wang et al., 2021] [Xia et al., 2020]
[Xia et al., 2021] [Xia and Durme, 2021] [Ye et al., 2021]
[Yoo et al., 2021] [Zha et al., 2021] [Zhao et al., 2021]
[Zhong et al., 2021a] [Zhong et al., 2021b]

Soft patterns

- Example for textual entailment:

discrete	$\underline{\text{P}}$. Question: $\underline{\text{H}}$? Answer: MASK .
soft	$\underline{\text{P}}$. $\underline{\text{H}}$? $v_1, \ldots, v_n$ MASK .

  $\underline{\text{P}}$ = premise, $\underline{\text{H}}$ = hypothesis

- $v_1, \ldots, v_n$ are trainable embeddings.
- Soft patterns can outperform discrete patterns.

Few-shot learning

Few-shot learning

- Dev sets in few-shot learning:
  Do they make sense?

Few-shot learning

- Dev sets in few-shot learning:
  Do they make sense?
- "True Few-Shot Learning with Language Models" (Perez, Kiela, Cho): cross-validation / minimum description length don't work.

Few-shot learning

- Dev sets in few-shot learning:
  Do they make sense?
- "True Few-Shot Learning with Language Models" (Perez, Kiela, Cho): cross-validation / minimum description length don't work.
- Single-pattern vs. ensemble

Few-shot learning

- Dev sets in few-shot learning:
  Do they make sense?
- "True Few-Shot Learning with Language Models" (Perez, Kiela, Cho): cross-validation / minimum description length don't work.
- Single-pattern vs. ensemble
- Variance: how to assess it?

Take the human out of the loop?

Take the human out of the loop?

- PET: system creator designs patterns/verbalizers.

Take the human out of the loop?

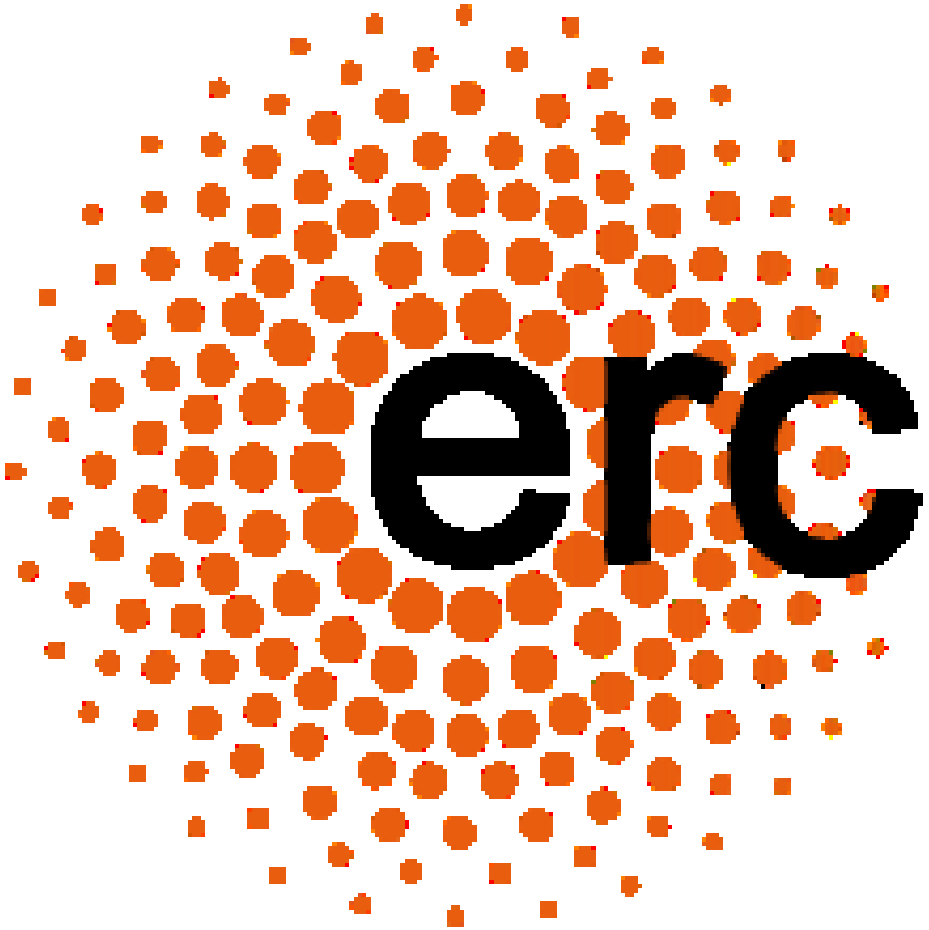
- PET: system creator designs patterns/verbalizers.
- Why can't we automate this?

Take the human out of the loop?

- PET: system creator designs patterns/verbalizers.
- Why can't we automate this?
- Because: inductive bias, domain knowledge, limited amount of information in the few shots, human learning

Summary

- Task descriptions:
  A new architectural element in machine learning
- If we want to emulate human learning,
  then we should use task descriptions.
- Good results with small models on SuperGLUE
- May even be useful for addressing bias in NLP


erc

Eyal Ben-David, Nadav Oved, and Roi Reichart. Pada: A prompt-based autoregressive approach for adaptation to unseen domains. *arXiv preprint arXiv:2102.12206*, 2021.

G. Betz and Kyle Richardson. Critical thinking for language models. *ArXiv*, abs/2009.07185, 2020.

G. Betz, Kyle Richardson, and Christian Voigt. Thinking aloud: Dynamic context generation improves zero-shot reasoning performance of gpt-2. *ArXiv*, abs/2103.13033, 2021.

Samuel R. Bowman and George E. Dahl. What will it take to fix benchmarking in natural language understanding? *ArXiv*, abs/2104.02145, 2021.

Xiang Chen, Xin Xie, Ningyu Zhang, Jiahuan Yan, Shumin Deng, Chuanqi Tan, Fei Huang, Luo Si, and Huajun Chen. Adaprompt: Adaptive prompt-based finetuning for relation extraction, 2021.

Xinyi Chen, J. Xu, and A. Wang. Label representations in modeling classification as text generation. In *AACL*, 2020.

Yihe Dong, Jean-Baptiste Cordonnier, and Andreas Loukas.

Attention is not all you need: Pure attention loses rank doubly exponentially with depth. *ArXiv*, abs/2103.03404, 2021.

Zhengxiao Du, Yujie Qian, Xiao Liu, Ming Ding, Jiezhong Qiu, Zhilin Yang, and Jie Tang. All nlp tasks are generation tasks: A general pretraining framework. *ArXiv*, abs/2103.10360, 2021.

Alexandre Duval, Thomas Lamson, Gael de Leseleuc de Kerouara, and Matthias Gallé. Breaking writer's block: Low-cost fine-tuning of natural language generation models. In *EACL*, 2021.

Tianyu Gao, Adam Fisch, and Danqi Chen. Making pre-trained language models better few-shot learners. *CoRR*, abs/2012.15723, 2020. URL `https://arxiv.org/abs/2012.15723`.

Demi Guo, Alexander M. Rush, and Y. Kim. Parameter-efficient transfer learning with diff pruning. *ArXiv*, abs/2012.07463, 2020.

Karen Hambardzumyan, Hrant Khachatrian, and Jonathan May. Warp: Word-level adversarial reprogramming, 2021.

Xu Han, Weilin Zhao, Ning Ding, Zhiyuan Liu, and Maosong Sun. Ptr: Prompt tuning with rules for text classification. 2021.

Adi Haviv, Jonathan Berant, and Amir Globerson. BERTese: Learning to speak to BERT. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 3618–3623, Online, April 2021. Association for Computational Linguistics. URL `https://www.aclweb.org/anthology/2021.eacl-main.316`.

Michael A. Hedderich, Lukas Lange, Heike Adel, Jannik Strotgen, and D. Klakow. A survey on recent approaches for natural language processing in low-resource scenarios. *ArXiv*, abs/2010.12309, 2021.

Jonathan Herzig, Peter Shaw, Ming-Wei Chang, Kelvin Guu, Panupong Pasupat, and Yuan Zhang. Unlocking compositional generalization in pre-trained models using intermediate representations. *ArXiv*, abs/2104.07478, 2021.

Ari Holtzman, Peter West, Vered Schwartz, Yejin Choi, and Luke Zettlemoyer. Surface form competition: Why the highest probability answer isn't always right. *ArXiv*, abs/2104.08315, 2021.

Jiaxin Ju, Jheng-Hong Yang, and Chuan-Ju Wang. Text-to-text multi-view learning for passage re-ranking. *ArXiv*, abs/2104.14133, 2021.

Masahiro Kaneko and Danushka Bollegala. Unmasking the mask - evaluating social biases in masked language models. *ArXiv*, abs/2104.07496, 2021.

Jack Lanchantin, Arshdeep Sekhon, Clint Miller, and Yanjun Qi. Transfer learning with motiftransformers for predicting protein-protein interactions between a novel virus and humans. *bioRxiv*, 2020.

Kenton Lee, Kelvin Guu, Luheng He, Timothy Dozat, and Hyung Won Chung. Neural data augmentation via example extrapolation. *ArXiv*, abs/2102.01335, 2021.

Brian Lester, Rami Al-Rfou, and Noah Constant. The power of scale for parameter-efficient prompt tuning. *ArXiv*, abs/2104.08691, 2021.

Xiang Lisa Li and Percy Liang. Prefix-tuning: Optimizing continuous prompts for generation, 2021.

Yingjie Li and Cornelia Caragea. Target-aware data augmentation for stance detection. In *NAACL*, 2021.

Jimmy Lin, Rodrigo Nogueira, and Andrew Yates. Pretrained transformers for text ranking: Bert and beyond. *Proceedings of the 14th ACM International Conference on Web Search and Data Mining*, 2021.

Xiao Liu, Yanan Zheng, Zhengxiao Du, Ming Ding, Yujie Qian, Zhilin Yang, and Jie Tang. Gpt understands, too. *ArXiv*, abs/2103.10385, 2021.

J. Lu, Pinghua Gong, Jieping Ye, and C. Zhang. Learning from very few samples: A survey. *ArXiv*, abs/2009.02653, 2020.

Yao Lu, Max Bartolo, A. Moore, S. Riedel, and Pontus Stenetorp. Fantastically ordered prompts and where to find them:

Overcoming few-shot prompt order sensitivity. *ArXiv*, abs/2104.08786, 2021.

Swaroop Mishra, Daniel Khashabi, Chitta Baral, and Hanna Hajishirzi. Natural instructions: Benchmarking generalization to new tasks from natural language instructions. *ArXiv*, abs/2104.08773, 2021.

Joseph D. Monaco, Kanaka Rajan, and Grace M. Hwang. A brain basis of dynamical intelligence for ai and computational neuroscience. *ArXiv*, abs/2105.07284, 2021.

Shikhar Murty, T. Hashimoto, and Christopher D. Manning. Dreca: A general task augmentation strategy for few-shot natural language inference. In *NAACL*, 2021.

Giovanni Paolini, Ben Athiwaratkun, Jason Krone, Jie Ma, A. Achille, Rishita Anubhai, C. D. Santos, Bing Xiang, and Stefano Soatto. Structured prediction as translation between augmented natural languages. *ArXiv*, abs/2101.05779, 2021.

Ethan Perez, Douwe Kiela, and Kyunghyun Cho. True few-shot learning with language models. 2021.

Guanghui Qin and Jason Eisner. Learning how to ask: Querying lms with mixtures of soft prompts. *CoRR*, abs/2104.06599, 2021. URL `https://arxiv.org/abs/2104.06599`.

Natraj Raman, Armineh Nourbakhsh, S. Shah, and M. Veloso. Robust document representations using latent topics and metadata. *ArXiv*, abs/2010.12681, 2020.

Nils Rethmeier and Isabelle Augenstein. Long-tail zero and few-shot learning via contrastive pretraining on and for small data. *ArXiv*, abs/2010.01061, 2020.

Mar'ia Virginia Sabando, I. Ponzoni, E. Milios, and Axel J. Soto. Using molecular embeddings in qsar modeling: Does it make a difference? *ArXiv*, abs/2104.02604, 2021.

Gabriele Sarti. Umberto-mtsa @ accompl-it: Improving complexity and acceptability prediction with multi-task learning on self-supervised annotations. *ArXiv*, abs/2011.05197, 2020.

Nikunj Saunshi, Sadhika Malladi, and Sanjeev Arora. A mathematical exploration of why language models help solve downstream tasks. *ArXiv*, abs/2010.03648, 2020.

Teven Le Scao and Alexander M. Rush. How many data points is a prompt worth?, 2021.

Richard Shin, C. H. Lin, Sam Thomson, Charles Chen, Subhro Roy, Emmanouil Antonios Platanios, Adam Pauls, D. Klein, J. Eisner, and Benjamin Van Durme. Constrained language models yield few-shot semantic parsers. *ArXiv*, abs/2104.08768, 2021.

Taylor Shin, Yasaman Razeghi, Robert L. Logan IV au2, Eric Wallace, and Sameer Singh. Autoprompt: Eliciting knowledge from language models with automatically generated prompts, 2020.

Connor Shorten, T. Khoshgoftaar, and B. Furht. Deep learning applications for covid-19. *Journal of Big Data*, 8, 2021.

Derek Tam, Rakesh R Menon, Mohit Bansal, Shashank Srivastava, and Colin Raffel. Improving and simplifying pattern exploiting training, 2021.

N. Tang, Ju Fan, Fangyi Li, Jianhong Tu, Xiaoyong Du, Guoliang Li, S. Madden, and M. Ouzzani. Rpt: Relational pre-trained

transformer is almost all you need towards democratizing data preparation. *Proc. VLDB Endow.*, 14:1254–1261, 2021.

Sinong Wang, Han Fang, Madian Khabsa, Hanzi Mao, and Hao Ma. Entailment as few-shot learner. *ArXiv*, abs/2104.14690, 2021.

Yutong Wang and C. Scott. An exact solver for the weston-watkins svm subproblem. *ArXiv*, abs/2102.05640, 2021.

Congyin Xia, Wenpeng Yin, Yihao Feng, and Philip Yu. Incremental few-shot text classification with multi-round new classes: Formulation, dataset and system. *ArXiv*, abs/2104.11882, 2021.

Patrick Xia and Benjamin Van Durme. Moving on from ontonotes: Coreference resolution model transfer. *ArXiv*, abs/2104.08457, 2021.

Patrick Xia, Shijie Wu, and Benjamin Van Durme. Which *bert? a survey organizing contextualized encoders. In *EMNLP*, 2020.

Qinyuan Ye, Bill Yuchen Lin, and Xiang Ren. Crossfit: A few-shot

learning challenge for cross-task generalization in nlp. *ArXiv*, abs/2104.08835, 2021.

Kang Min Yoo, Dongju Park, Jaewook Kang, Sang-Woo Lee, and Woomyeong Park. Gpt3mix: Leveraging large-scale language models for text augmentation. *ArXiv*, abs/2104.08826, 2021.

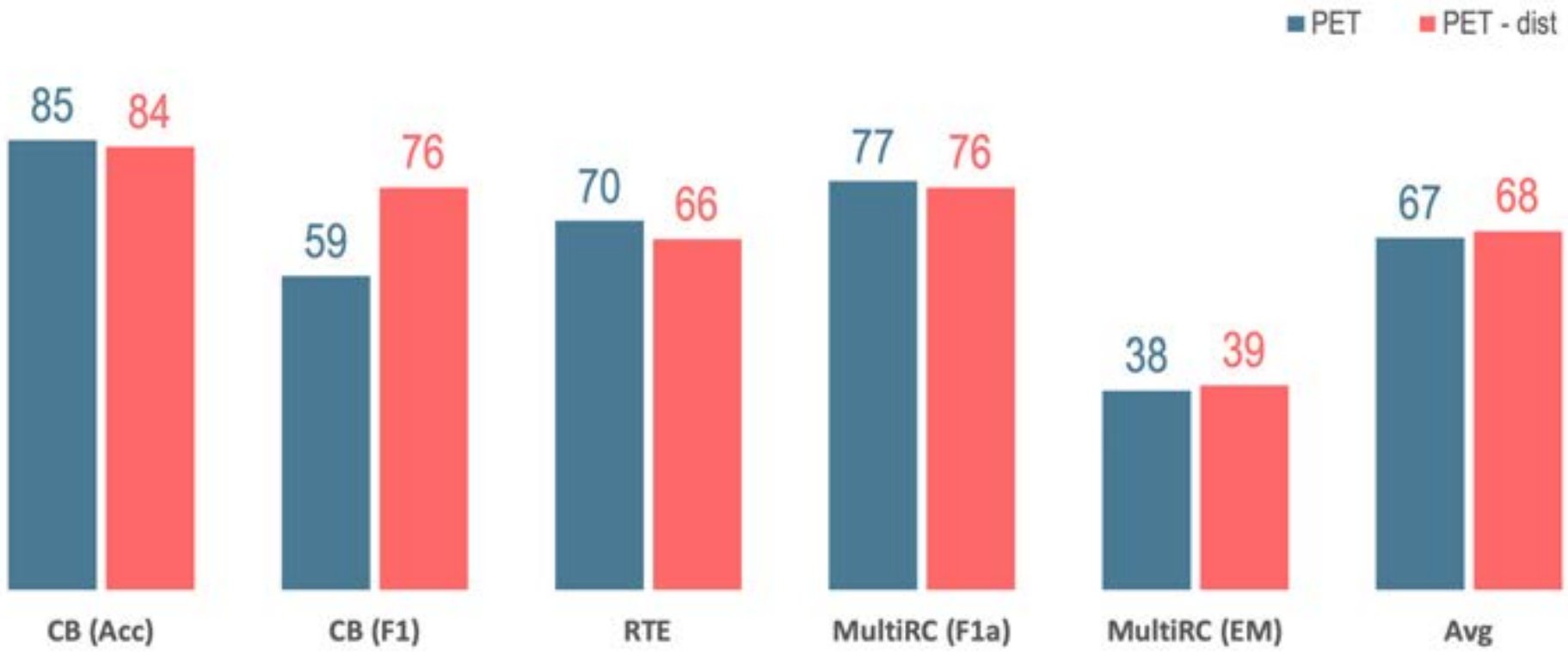
H. Zha, Zhiyu Chen, and X. Yan. Inductive relation prediction by bert. *ArXiv*, abs/2103.07102, 2021.

Tony Zhao, Eric Wallace, Shi Feng, D. Klein, and Sameer Singh. Calibrate before use: Improving few-shot performance of language models. *ArXiv*, abs/2102.09690, 2021.

Ruiqi Zhong, Kristy Lee, Zheng Zhang, and D. Klein. Meta-tuning language models to answer prompts better. *ArXiv*, abs/2104.04670, 2021a.

Zexuan Zhong, Dan Friedman, and Danqi Chen. Factual probing is [mask]: Learning vs. learning to recall, 2021b.

Effect of (not) using unlabeled data



ALBERT xxlarge

Distillation creates single model from pattern-specific individual models

Distillation:

- Use individual models to label an unlabeled dataset $\mathcal{T}$
- Aggregrate scores to label $\mathcal{T}$
- Train final PET model on T