The Appeal of Parameter-efficient Transfer Learning

Tu Vu

June 7, 2022

Agenda

Background on Transfer Learning & Prompt Tuning

SPoT: Better Frozen Model Adaptation through Soft Prompt Transfer
ACL 2022

Overcoming Catastrophic Forgetting in Zero-Shot Cross-Lingual Generation
EMNLP 2022 submission

Future work

The dominant transfer learning paradigm

Transfer Learning

- pre-train a model on a task before fine-tuning it on another (downstream) task

Language Model (LM) Pre-training & Fine-tuning

Unsupervised Pre-training

original text: Thank you for attending my talk today. I hope you enjoy it!

input: Thank you MASK my talk MASK I hope you MASK

target: for attending, today, enjoy it!

adapting

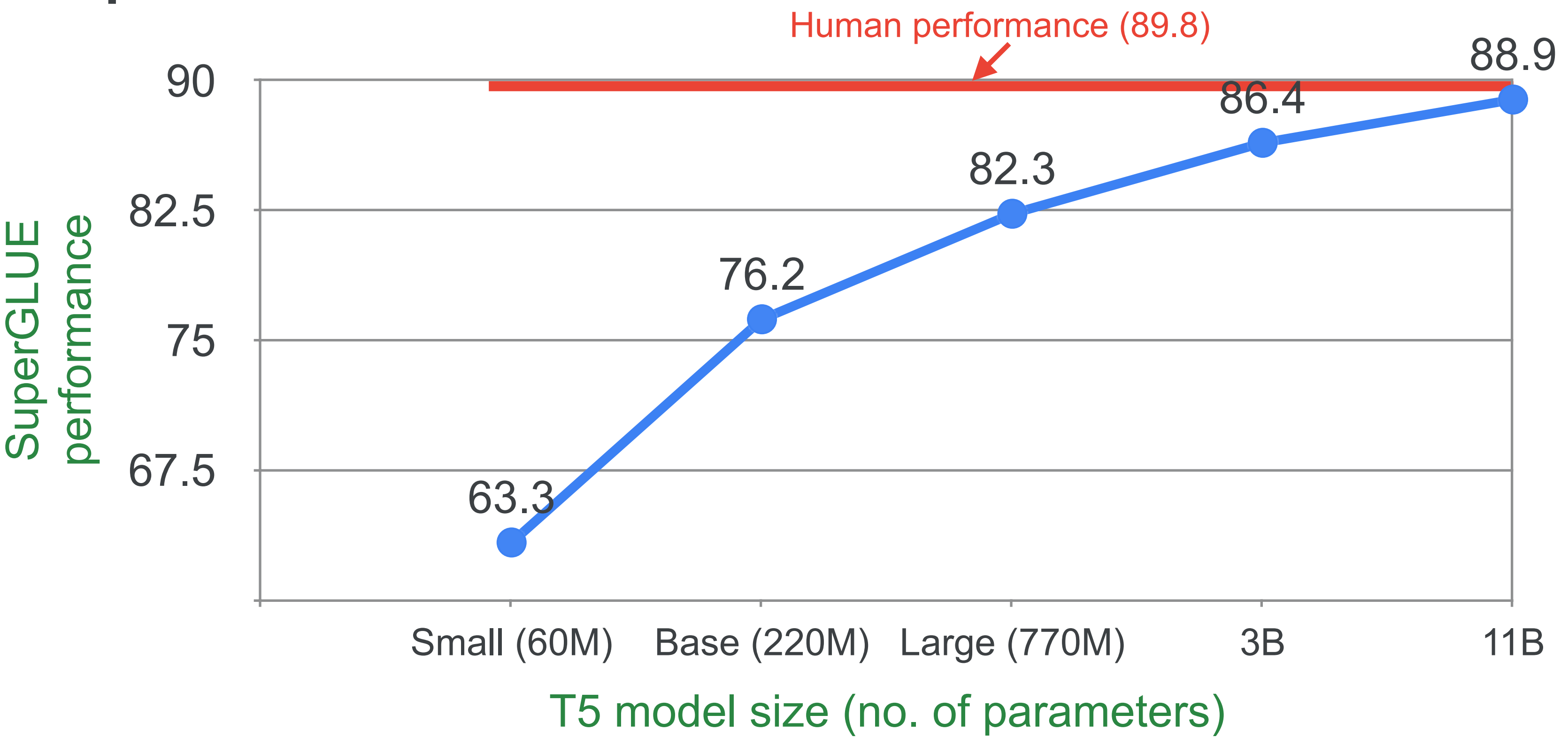
Supervised Fine-tuning

input: This movie is absolutely INCREDIBLE! Seriously one of my all time favorites.

target: positive

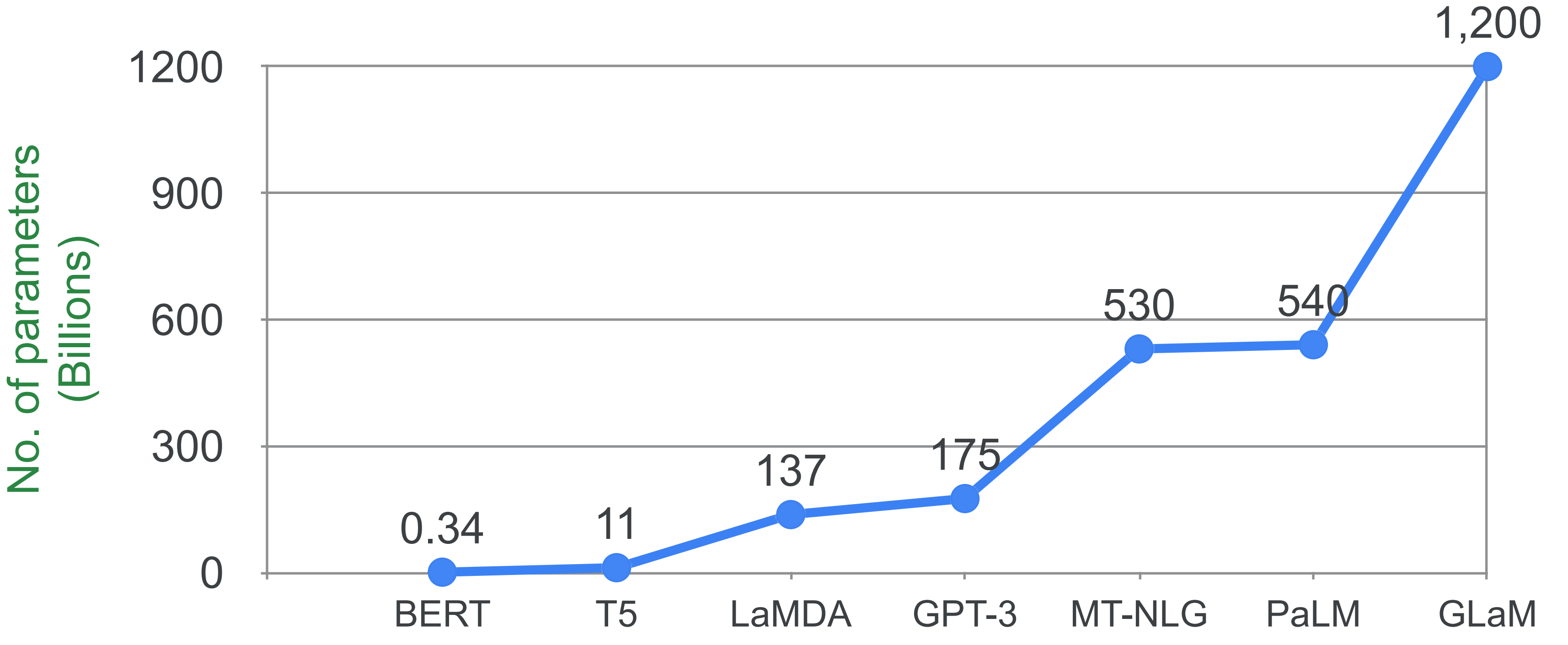
(Inspired by Figure 2 in Raffel et al. (2020)
& Slide 3 in Raffel et al. (2021))

Scaling up the model size is a key ingredient for achieving the best performance



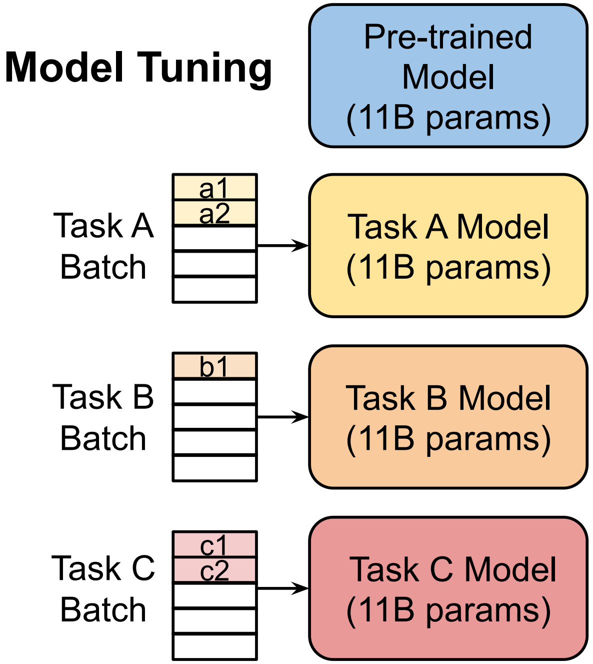
Numbers taken from [Raffel et al. (2020)]

The trend has continued to push the boundaries of possibility in NLP



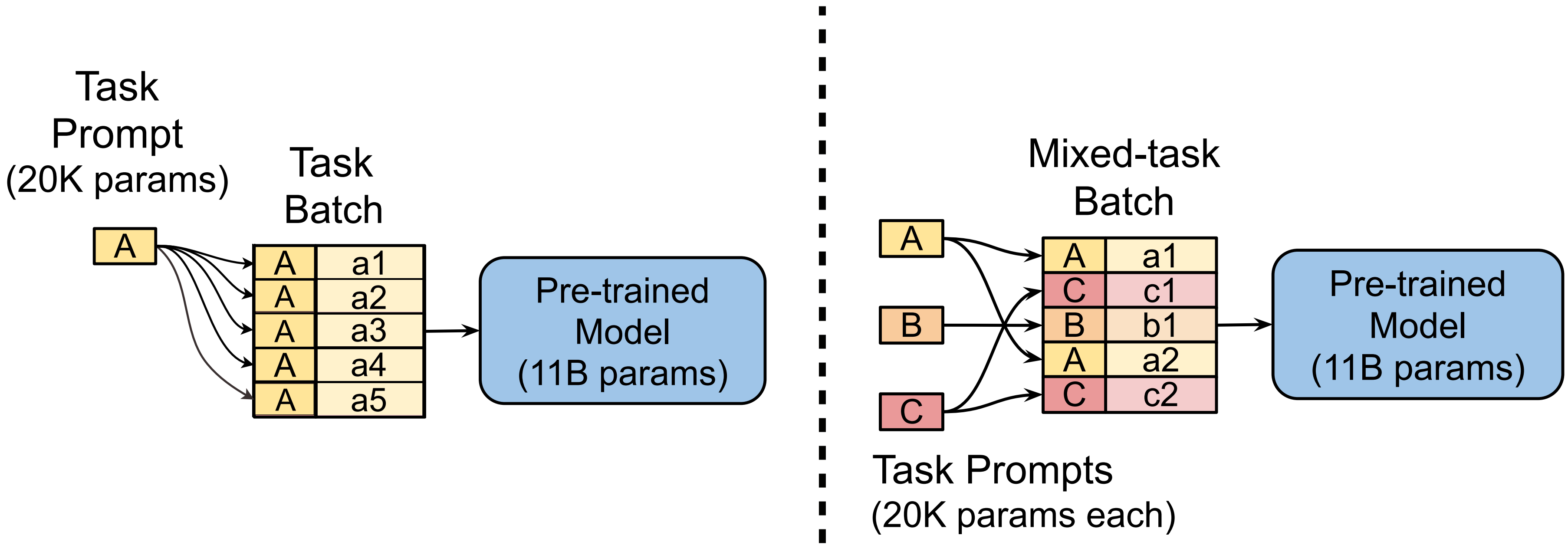
When BERT becomes small

Drawback: Large-scale pre-trained language models are costly to share and serve



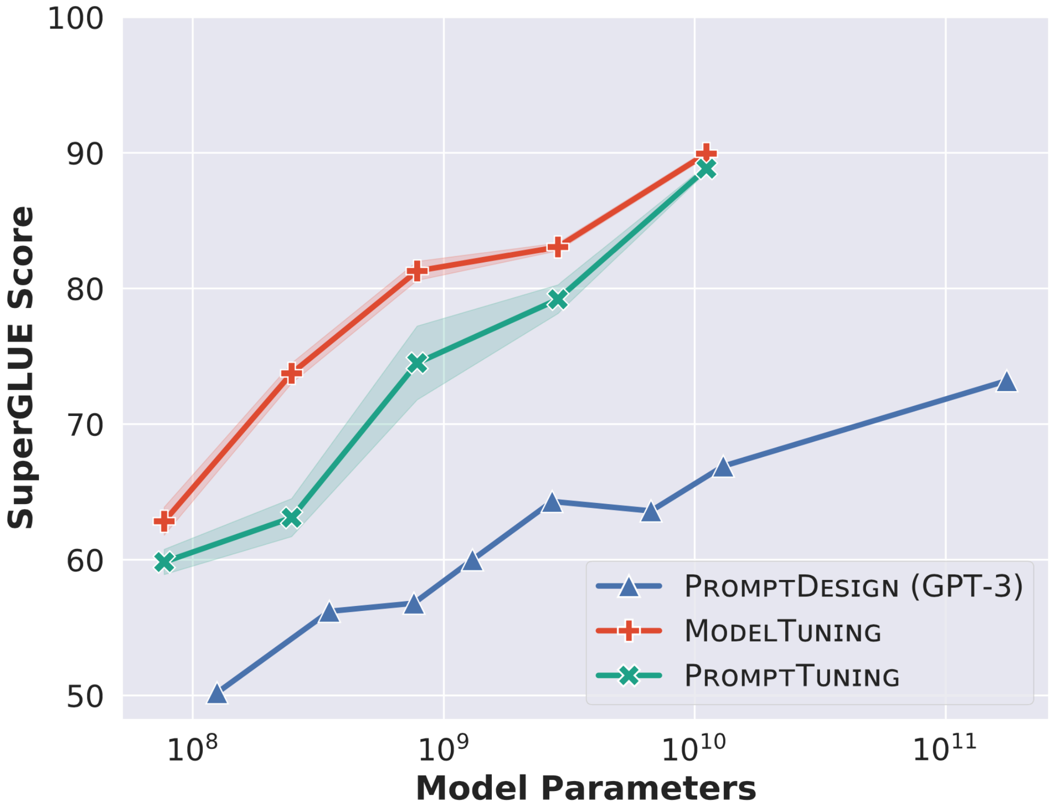
[Lester et al., 2021]

Prompt Tuning (Lester et al., 2021) to the rescue!



Lester et al., 2021

Prompt Tuning becomes competitive with Model Tuning as model capacity increases



Other parameter-efficient tuning methods

differ in what they tune during adaptation

- a small number of model parameters (**BiTFiT**; Zakhen et al., 2019)
- added task-specific modules, e.g.,
    - prefixes (**Prefix Tuning**; Li and Liang, 2021)
    - adapters (Houlsby et al., 2019)
    - low-rank structures (**LoRA**; Hu et al., 2022)
    - rescaling vectors (**(IA)^3**; Liu et al., 2022)

Advantages of Prompt Tuning over other parameter-efficient tuning methods

Parameter efficiency

- < 0.01% task-specific parameters

Simplicity

- no model architecture modifications

Mixed-task inference

Improved performance with scale

Interpretability

- could possibly be interpreted as natural language instructions

Research questions

R1: How to facilitate transfer learning as model capacity increases?

➡ **SPoT**

R2: Can current transfer learning methods extend successfully to a zero-shot cross-lingual transfer setting?

➡ **xGen**

Research questions

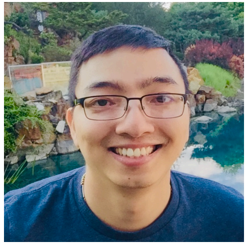
R1: How to facilitate transfer learning as model capacity increases?

➡ **SPoT**

R2: Can current transfer learning methods extend successfully to a zero-shot cross-lingual transfer setting?

➡ **xGen**

SPoT: Better Frozen Model Adaptation through Soft Prompt Transfer

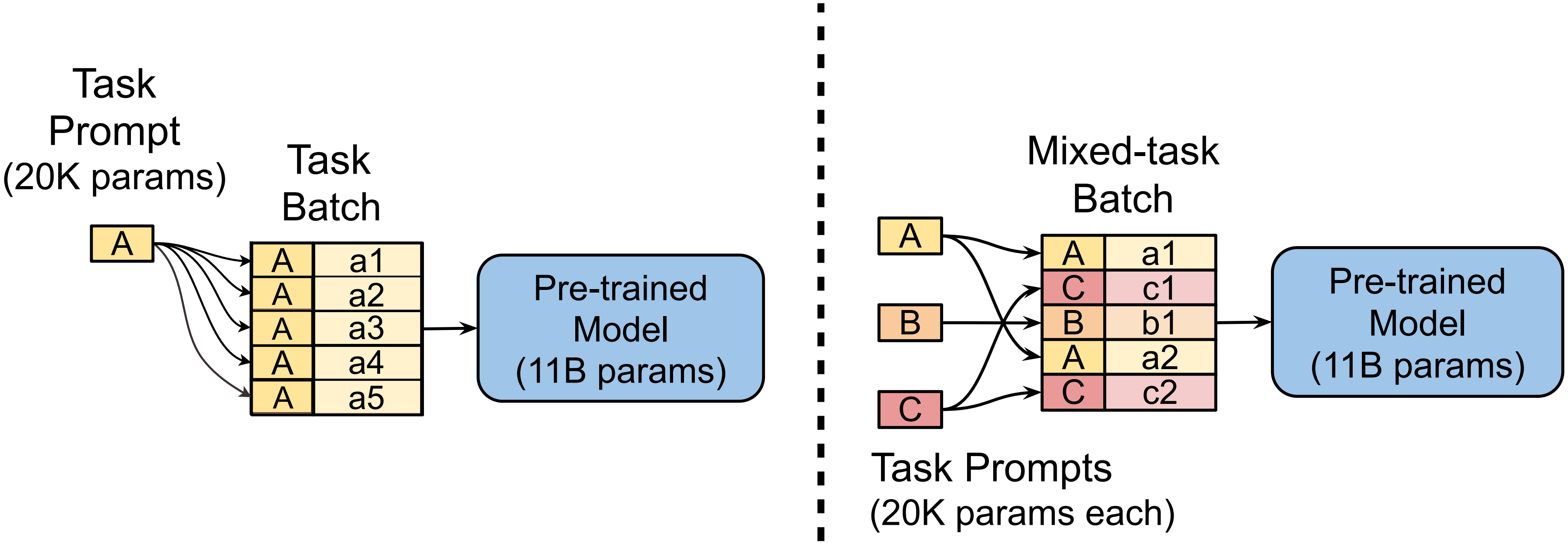




Tu Vu[1,2] Brian Lester[1] Noah Constant[1] Rami Al-Rfou[1] Daniel Cer[1]

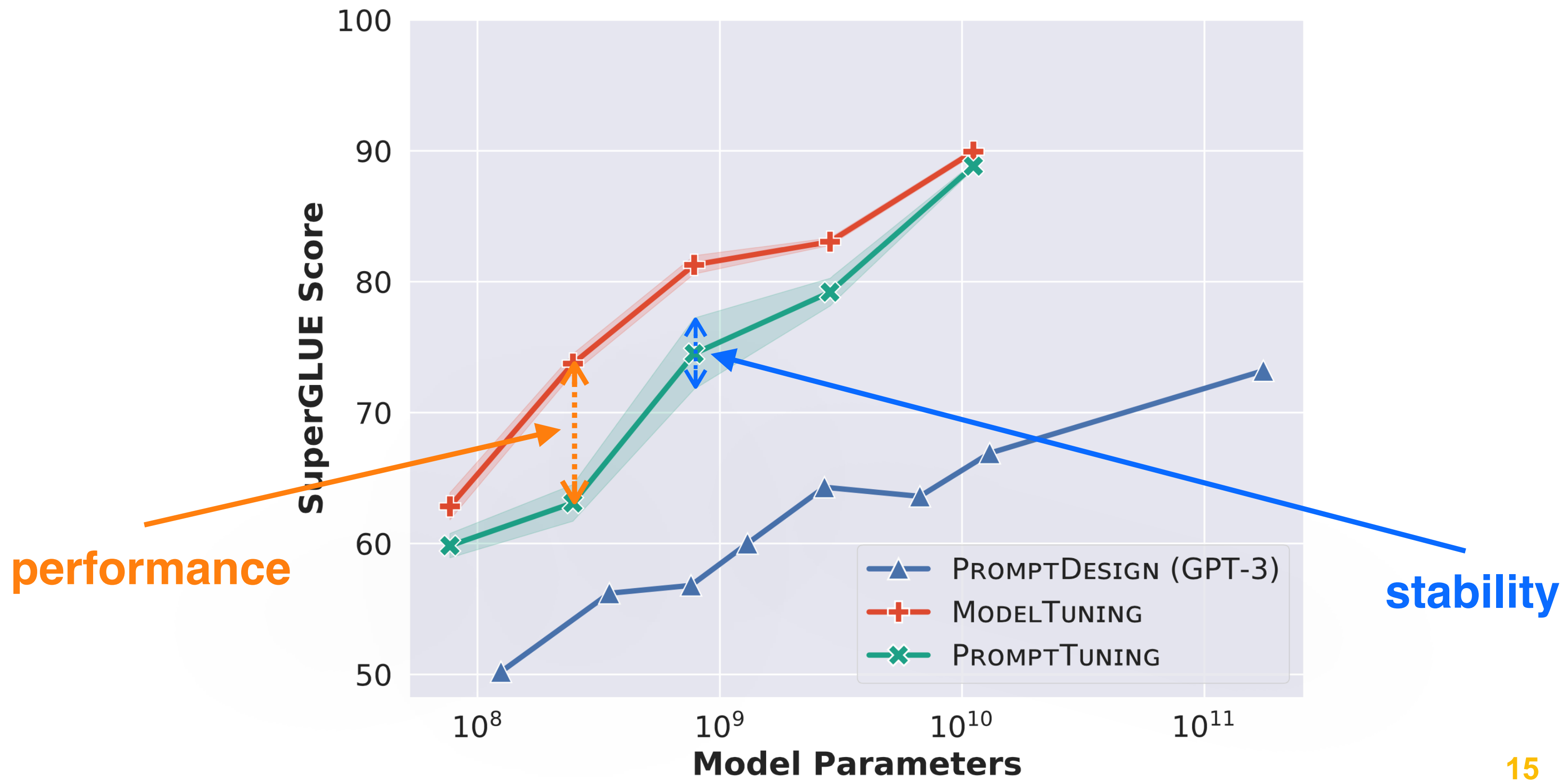
Google Research[1] UMass Amherst[2]

Parameter-efficient Prompt Tuning (Lester et al., 2021)

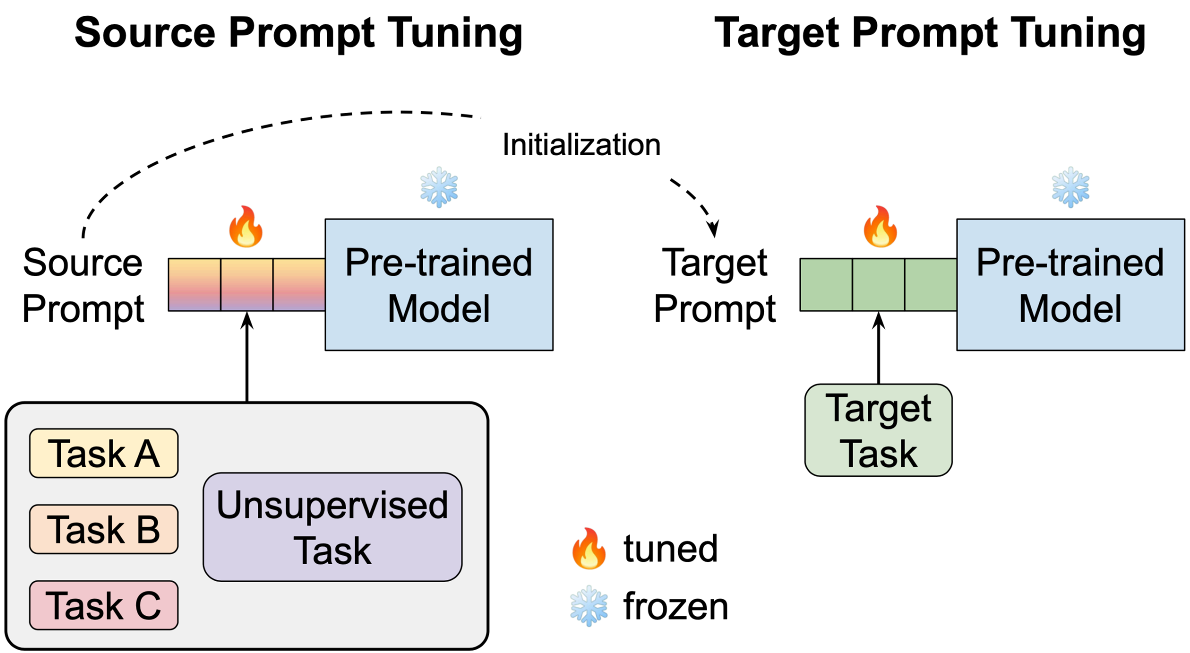


Lester et al., 2021

Significant headroom remains

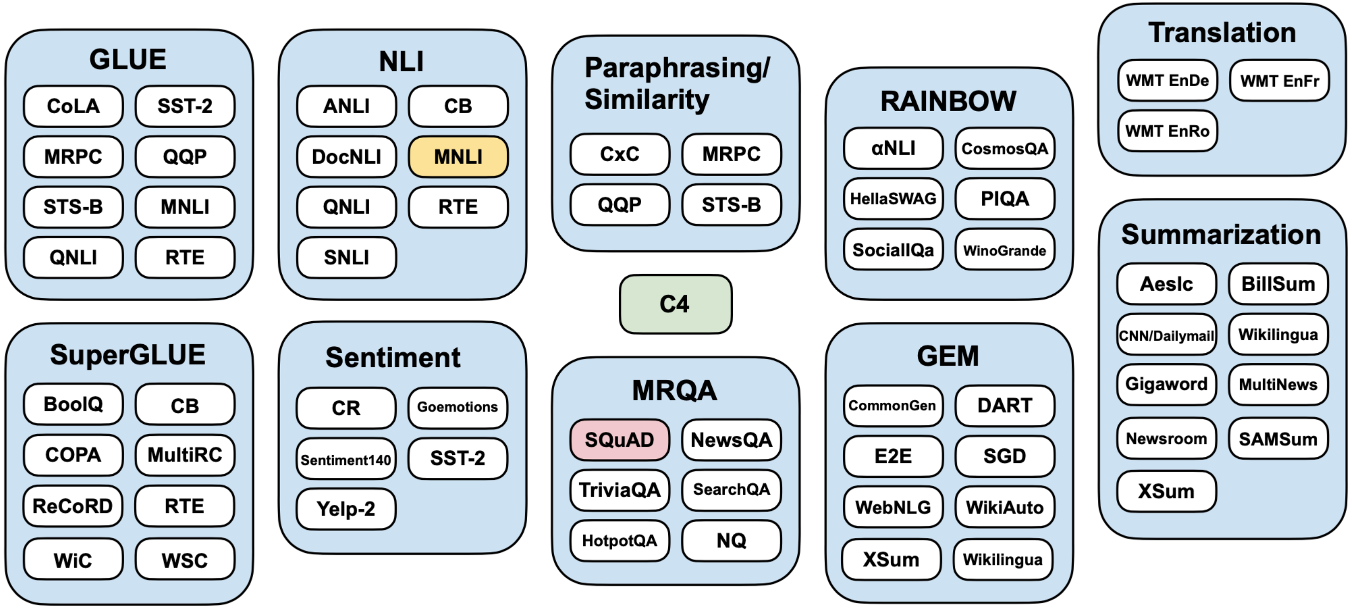


Our *generic* SPoT approach



We learn a single generic source prompt on one or more source tasks, which is then used to initialize the prompt for each target task.

Mixing datasets from different benchmarks / task families



Datasets used in our experiments. C4, MNLI, and SQuAD were all used by themselves as single source tasks in addition to being mixed in with other tasks.

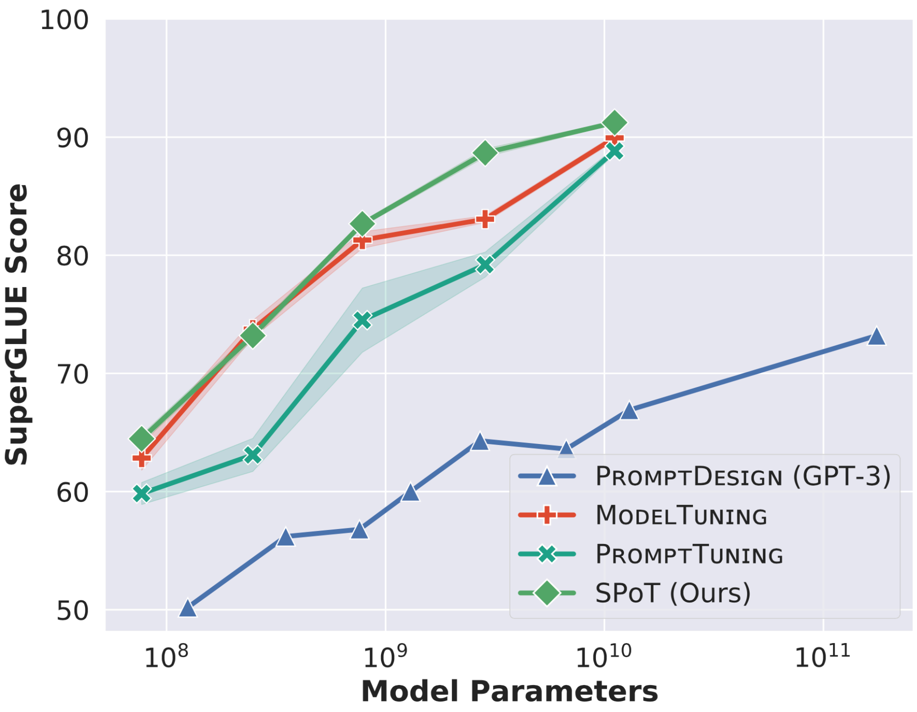
SPoT significantly improves performance and stability of Prompt Tuning

GLUE and SUPERGLUE results achieved by applying T5 BASE with different prompt tuning approaches. We report the mean and standard deviation (in the subscript) across three random seeds.

Method	GLUE	SUPERGLUE
BASELINE		
PROMPTTUNING	$81.2_{0.4}$	$66.6_{0.2}$
– longer tuning	$78.4_{1.7}$	$63.1_{1.1}$
SPOT *with different source mixtures*		
GLUE (*8 tasks*)	$\mathbf{82.8}_{0.2}$	$\mathbf{73.2}_{0.3}$
– longer tuning	$82.0_{0.2}$	$70.7_{0.4}$
C4	$82.0_{0.2}$	$67.7_{0.3}$
MNLI	$82.5_{0.0}$	$72.6_{0.8}$
SQUAD	$82.2_{0.1}$	$72.0_{0.4}$
SUPERGLUE (8 tasks)	$82.0_{0.1}$	$66.6_{0.2}$
NLI (*7 tasks*)	$82.6_{0.1}$	$71.4_{0.2}$
Paraphrasing/similarity (*4 tasks*)	$82.2_{0.1}$	$69.7_{0.5}$
Sentiment (*5 tasks*)	$81.1_{0.2}$	$68.6_{0.1}$
MRQA (*6 tasks*)	$81.8_{0.2}$	$68.4_{0.2}$
RAINBOW (*6 tasks*)	$80.3_{0.6}$	$64.0_{0.4}$
Translation (*3 tasks*)	$82.4_{0.2}$	$65.3_{0.1}$
Summarization (*9 tasks*)	$80.9_{0.3}$	$67.1_{1.0}$
GEM (*8 tasks*)	$81.9_{0.2}$	$70.5_{0.5}$
All (C4 + 55 supervised tasks)	$81.8_{0.2}$	$67.9_{0.9}$

SPoT helps close the gap with Model Tuning across model sizes

Our SPoT approach—which transfers a prompt learned from a mixture of source tasks (here, GLUE) onto target tasks—outperforms vanilla PromtTuning and GPT-3 on SuperGLUE by a large margin, matching or outperforming ModelTuning across all model sizes.



SPoT is competitive with methods that tune billions of parameters

	Model	Total parameters	Tuned parameters	SCORE
Top-7 submissions	ST-MOE-32B	269B	269B	**91.2**
	TURING NLR V5	5.4B	5.4B	90.9
	ERNIE 3.0	12B	12B	90.6
	T5 + UDG	11B	11B	90.4
	DEBERTA / TURINGNLRV4	3.1B	3.1B	90.3
	HUMAN BASELINES	-	-	89.8
	T5	11B	11B	89.3
Parameter-efficient adaptation	FROZEN T5 1.1 + SPOT	11B	410K	**89.2**
	GPT-3 FEW-SHOT	175B	0	71.8
	WARP FEW-SHOT	223M	25K	48.7
	CBOW	15M	33K	44.5

SUPERGLUE results of our SPOT XXL submission and competitors from the leaderboard as of 2022/02/09.

A large-scale study on task transferability in the context of prompt tuning

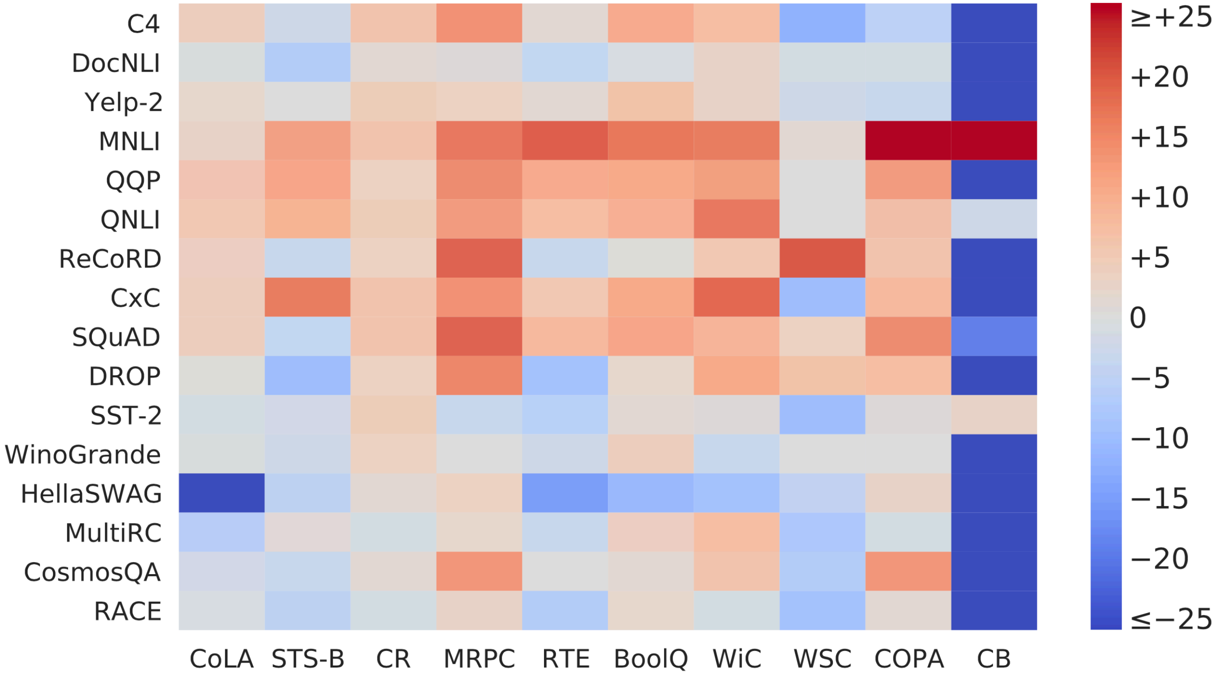
26 NLP tasks

- 16 source tasks, 10 target tasks,
  160 source-target combinations of tasks
- covering various task types

Name	Task type	|Train|
16 source tasks		
C4	language modeling	365M
DocNLI	NLI	942K
Yelp-2	sentiment analysis	560K
MNLI	NLI	393K
QQP	paraphrase detection	364K
QNLI	NLI	105K
ReCoRD	QA	101K
CxC	semantic similarity	88K
SQuAD	QA	88K
DROP	QA	77K
SST-2	sentiment analysis	67K
WinoGrande	commonsense reasoning	40K
HellaSWAG	commonsense reasoning	40K
MultiRC	QA	27K
CosmosQA	commonsense reasoning	25K
RACE	QA	25K
10 target tasks		
BoolQ	QA	9K
CoLA	grammatical acceptability	9K
STS-B	semantic similarity	6K
WiC	word sense disambiguation	5K
CR	sentiment analysis	4K
MRPC	paraphrase detection	4K
RTE	NLI	2K
WSC	coreference resolution	554
COPA	QA	400
CB	NLI	250

Tasks used in our task transferability experiments, sorted by training dataset size.

Many tasks can benefit each other via prompt transfer



A heatmap of our task transferability results. Each cell shows the relative error reduction on the target task of the transferred prompt from the associated source task (row) to the associated target task (column).

Measuring task similarity through prompt similarity

Cosine Similarity of Average Tokens

- cosine similarity between the average pooled representations of the prompt tokens:

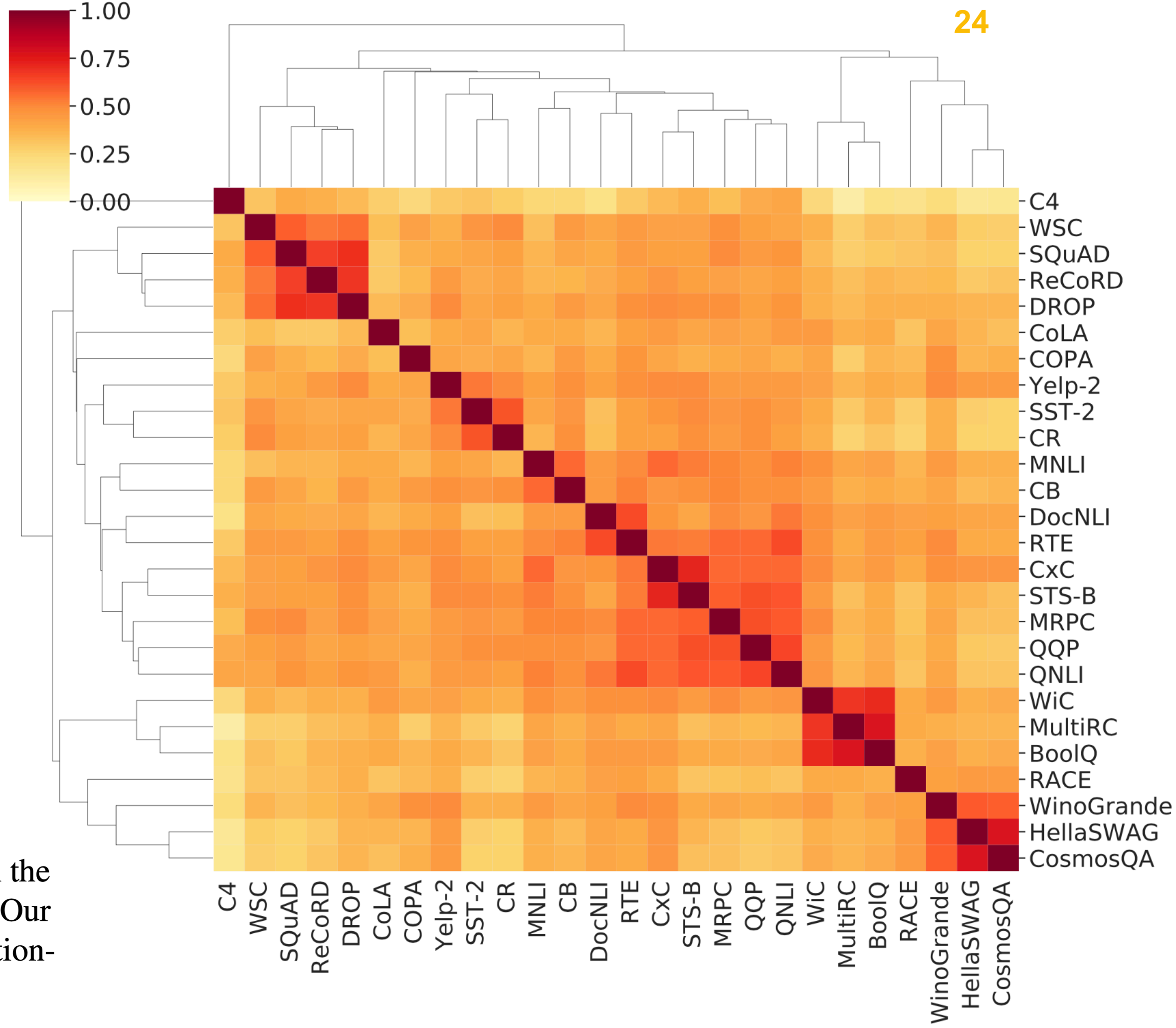
$$sim(t^1, t^2) = cos(\frac{1}{\mathcal{L}} \sum_i \boldsymbol{e}_i^1, \frac{1}{\mathcal{L}} \sum_j \boldsymbol{e}_j^2)$$

Per-token Average Cosine Similarity

- average cosine similarity between every prompt token pair:

$$sim(t^1, t^2) = \frac{1}{\mathcal{L}^2} \sum_i \sum_j cos(\boldsymbol{e}_i^1, \boldsymbol{e}_j^2)$$

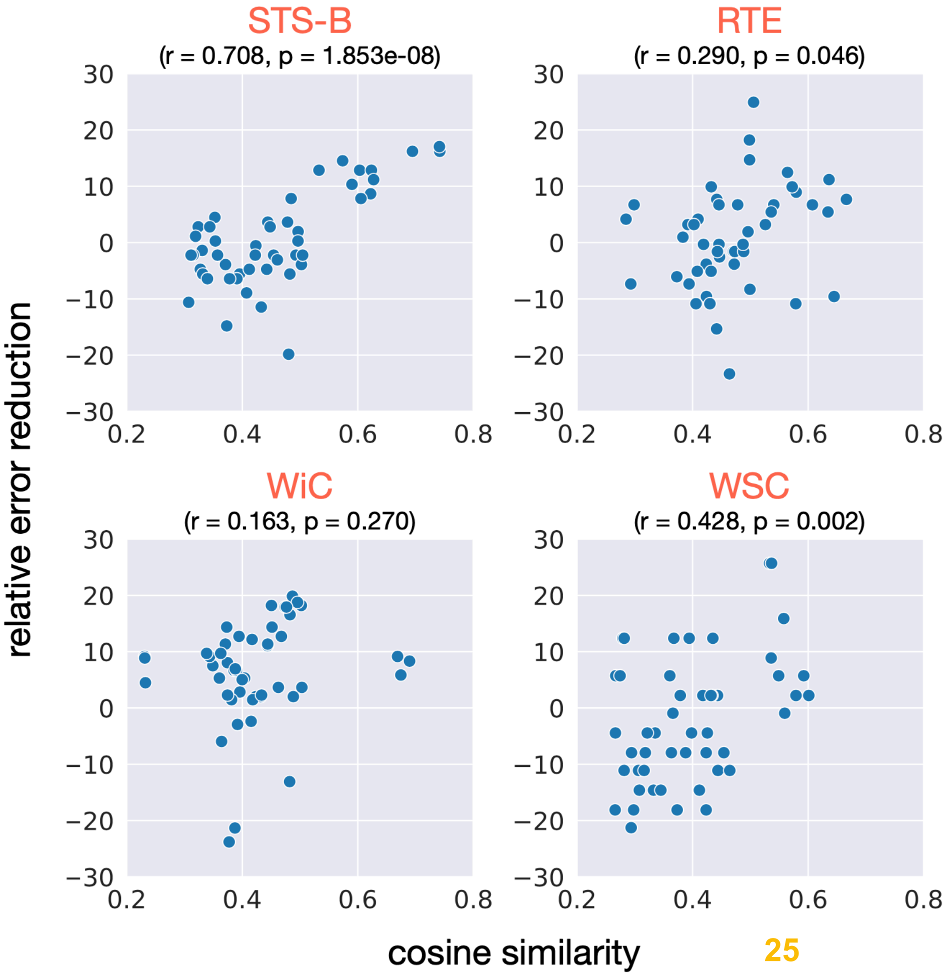
Prompt-based task embeddings capture task relationships



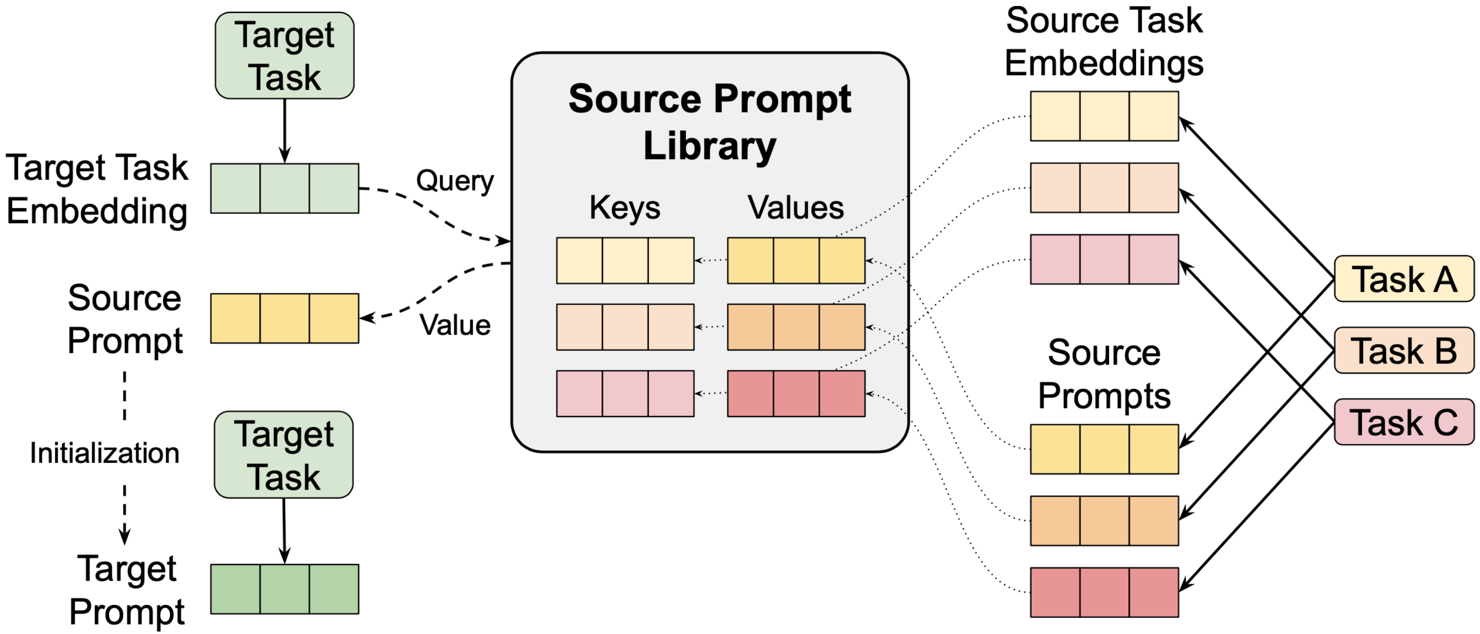
A clustered heatmap of cosine similarities between the task embeddings of the 26 NLP tasks we study. Our prompt-based task embeddings capture task relationships: similar tasks cluster together.

Correlation between task similarity & task transferability

Correlation between task similarity and task transferability. Each point represents a source prompt. The x-axis shows the cosine similarity between the associated source and target task embeddings, averaged over three runs for the target task (orange title). The y-axis measures the relative error reduction on the target task achieved by each source prompt. We include the Pearson correlation coefficient (r) and p-value.



Our *targeted* SPoT approach



We learn separate prompts for various source tasks, saving early checkpoints as task embeddings and best checkpoints as source prompts. These form the keys and values of our prompt library. Given a novel target task, a user: (i) computes a task embedding, (ii) retrieves an optimal source prompt, and (iii) trains a target prompt, initialized from the source prompt

Predicting task transferability via task similarity

Best of Top-k

- use the top-k source prompts individually

Top-k Weighted Average

- use a weighted average of the top-k source prompts

Top-k Multi-task Mixture

- pre-train the prompt on a mixture of source datasets whose prompts are in the top-k

Retrieving source tasks via task embeddings is helpful

Task embeddings provide an effective means of predicting and exploiting task transferability, eliminating 69% of the source task search space while keeping 90% of the best-case quality gain obtained by oracle selection.

Method	Avg. score
BASELINE	$74.7_{0.7}$
Brute-force search ($k = 48$)	
ORACLE	$80.7_{0.0}$
BEST OF TOP-k	
$k = 1$	$76.7_{0.7}$
$k = 3$	$77.5_{0.4}$
$k = 6$	$79.2_{0.1}$
$k = 9$	$79.5_{0.2}$
$k = 12$	$79.6_{0.1}$
$k = 15$	$80.0_{0.4}$
TOP-k WEIGHTED AVERAGE	
best $k = 3$	$76.6_{0.1}$
TOP-k MULTI-TASK MIXTURE	
best $k = 12$	$77.8_{0.1}$

Take-aways

1. Scale is not necessary for Prompt Tuning to match Model Tuning
   SPoT can match or beat Model Tuning across model sizes

2. Tasks can benefit each other via prompt transfer

3. Retrieving similar tasks via task embeddings is helpful

R1: How to facilitate transfer learning as model capacity increases?

➡ use parameter-efficient transfer methods, e.g., SPoT

Research questions

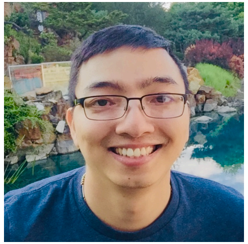
R1: How to facilitate transfer learning as model capacity increases?

➡ **SPoT**

R2: Can current transfer learning methods extend successfully to a zero-shot cross-lingual transfer setting?

➡ **xGen**

Overcoming Catastrophic Forgetting in Zero-Shot Cross-Lingual Generation



Tu Vu[1,2] Aditya Barua[1] Brian Lester[1]

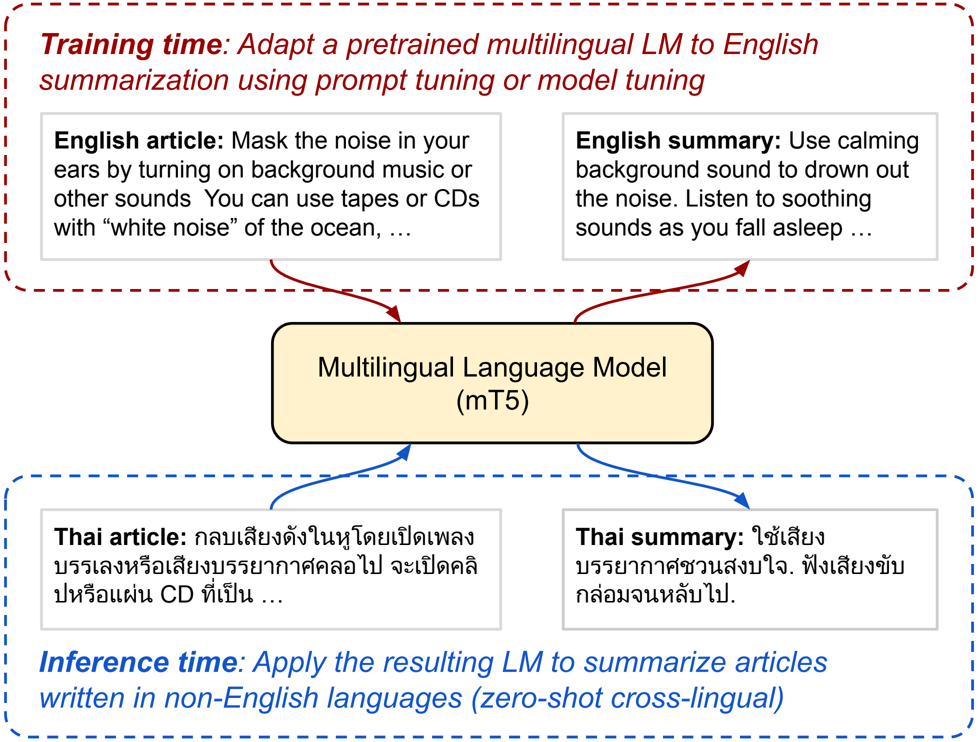


Daniel Cer[1] Mohit Iyyer[2] Noah Constant[1]

Google Research[1] UMass Amherst[2]

WikiLingua-0

A demonstration of WIKILINGUA-0, a challenging zero-shot cross-lingual generation (XGEN) task, which requires a model to learn a generative task from labeled data in one language (i.e., English), and then perform the equivalent task in another language at inference time.



Evaluation metrics

SP-Rouge

- SentencePiece Rouge that measures summarization quality

LID_*lang*

- the average confidence score given by cld3 when detecting the language *lang*

ASCII

- the average percentage of ASCII characters present in the text

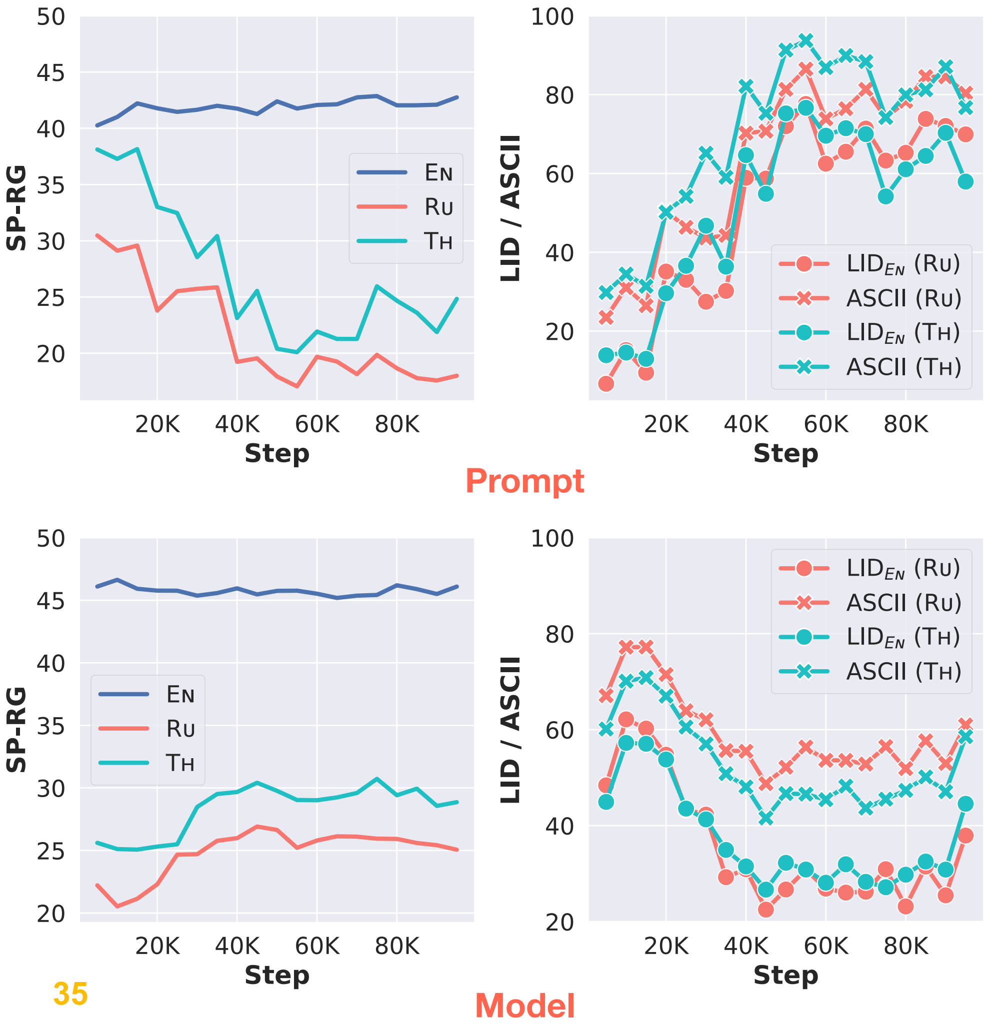
Prompt Tuning is preferred when there is a significant language shift at inference time

		TH		
Size	**Method**	SP-ROUGE	LID_{EN}	LID_{TH}
SMALL	PROMPT	14.9	**45.9**	**3.3**
SMALL	MODEL	**17.3**	78.1	0.1
BASE	PROMPT	17.3	**34.3**	**33.5**
BASE	MODEL	**17.9**	89.0	0.3
LARGE	PROMPT	24.7	**29.0**	**45.9**
LARGE	MODEL	**25.9**	36.5	35.4
XL	PROMPT	**33.2**	**19.8**	**66.0**
XL	MODEL	25.6	54.7	24.9
XXL	PROMPT	**37.4**	**13.5**	**75.5**
XXL	MODEL	30.1	32.7	16.8

PROMPTTUNING usually produces better results than MODELTUNING when there is a significant language shift at inference time.

Both Model Tuning and Prompt Tuning suffer from catastrophic forgetting

Learning curves showing how PROMPTTUNING (top) and MODELTUNING (bottom) progress in terms of summarization quality (left) and unwanted English output (right), at the XXL model size. Note, MODELTUNING quality is lower overall, and predictions contain high (>40%) levels of unwanted ASCII.



Training step	5K	10K	30K	50K
Prompt	กำหนดเป้าหมายของคุณ. แบ่งเป้าหมายของคุณออกเป็นเป้าหมายย่อยๆ. ตั้งเป้าหมายในทุกๆ วัน. ตั้งเป้าหมายในระยะยาว.	กำหนดเป้าหมายของคุณ. แบ่งเป้าหมายของคุณเป็นเป้าหมายย่อยๆ. ติดตามความก้าวหน้าของคุณ. ถามตัวเองว่า “ฉันสามารถทำอะไรในวันนี้เพื่อบรรลุเป้าหมายของฉัน? ฉันสามารถทำอะไรในอีก 3 สัปดาห์ข้างหน้าเพื่อบรรลุเป	Define your goal. Divide your goal into smaller goals. Keep track of your progress. Ask yourself what you can do today. Ask yourself what you can do in 3 weeks. Ask yourself what you can do in a year.	Establish a time frame for your goals. Divide your goals into smaller steps. Ask yourself what you can do today. Ask yourself what you can do in the next 3 weeks. Ask yourself what you can do in the n
Model	Set a time limit for your goal	Set a time limit for your goal	Be specific about your goals. Set benchmarks for yourself. Keep track of your progress. Ask yourself questions.	Be specific with your goals. Set benchmarks and routines to help you achieve your goals. Keep track of your progress. Ask yourself questions to help you stay on track

Bigger models are less prone to forget

		Th		
Size	**Method**	SP-Rouge	LID_{EN}	LID_{TH}
Small	Prompt	14.9	45.9	3.3
Base	Prompt	17.3	34.3	33.5
Large	Prompt	24.7	29.0	45.9
XL	Prompt	33.2	19.8	66.0
XXL	Prompt	**37.4**	**13.5**	**75.5**
Small	Model	17.3	78.1	0.1
Base	Model	17.9	89.0	0.3
Large	Model	25.9	36.5	**35.4**
XL	Model	25.6	54.7	24.9
XXL	Model	**30.1**	**32.7**	16.8

For both ModelTuning and PromptTuning, moving to larger model sizes mitigates catastrophic forgetting to a remarkable extent.

Too much capacity is harmful for Prompt Tuning

Size	Method	TH SP-ROUGE	TH LID_{EN}	TH LID_{TH}
BASE	PROMPT, L=1	19.2	**3.3**	**80.2**
	PROMPT, L=10	**21.0**	11.8	53.7
	PROMPT, L=100	17.3	34.3	33.5
	PROMPT, L=1000	16.3	47.5	18.9
XXL	PROMPT, L=1	36.4	**0.1**	**99.3**
	PROMPT, L=10	**41.2**	2.0	91.3
	PROMPT, L=100	37.4	13.5	75.5
	PROMPT, L=1000	37.8	7.4	81.7

An interesting "paradox of capacity" with regard to prompt length. One the one hand, greater capacity (in the form of longer prompts) clearly helps to better learn the summarization task. On the other hand, the greater the capacity to learn from English training data, the more the model forgets other languages. For each language and model size, we observe a "balance point" past which adding extra capacity becomes harmful.

Significant headroom remains

Size	Method	TH SP-ROUGE	TH LID$_{EN}$	TH LID$_{TH}$
XXL	PROMPT	37.4	13.5	75.5
XXL	PROMPT, TRANS-TEST	28.7	**0.0**	**100.0**
XXL	PROMPT, TRANS-TRAIN	37.1	**0.0**	**100.0**
XXL	PROMPT, SUP	**45.0**	0.1	99.6
XXL	MODEL	30.1	32.7	16.8
XXL	MODEL, TRANS-TEST	31.7	**0.0**	**100.0**
XXL	MODEL, TRANS-TRAIN	38.7	**0.0**	**100.0**
XXL	MODEL, SUP	**48.8**	**0.0**	99.9

When tuning the XXL model directly on supervised training data in each language (SUP), SP-ROUGE scores are much higher than our highest zero-shot results. For some languages, like Thai, the supervised baseline greatly exceeds any approach using machine translation (TRANS*).

Mitigating catastrophic forgetting

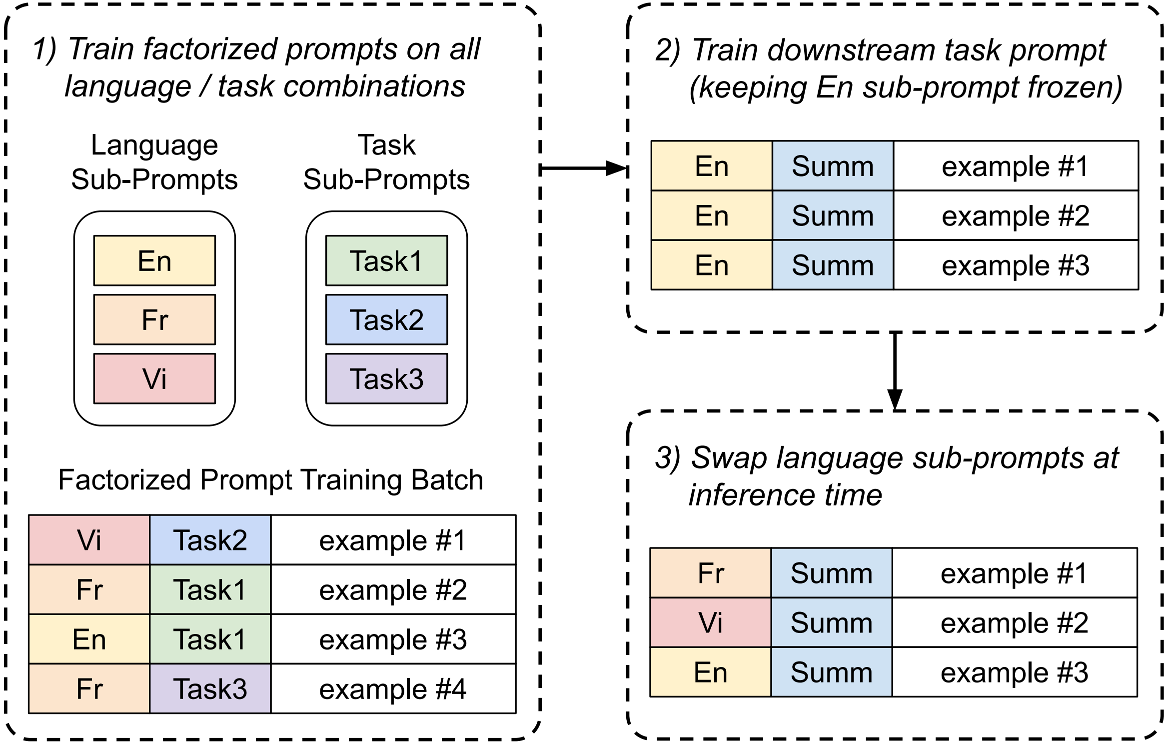
Mixing in unlabeled training data

- 1%: an unsupervised training task (i.e., span corruption) from the target language
- 99%: WikiLingua-0

Factorized prompts (specifically designed for Prompt Tuning)

- each prompt is decomposed into “task” and “language” sub-prompts that can be recombined in novel pairings (FP); inspired by MAD-X (Pfeiffer et al., 2021)

Factorized prompts



Our "factorized prompts" approach learns recomposable language and task sub-prompts by training on all language / task combinations from a set of 7 unsupervised language modeling tasks covering all 18 WIKILINGUA-0 languages.

Mixing in multilingual data prevents catastrophic forgetting

		Th		
Size	**Method**	SP-Rouge	LID_{EN}	LID_{TH}
Base	Prompt	17.3	34.3	33.5
Base	Prompt, Mix-Unsup	**20.9**	**4.1**	**76.9**
XXL	Prompt	**37.4**	**13.5**	**75.5**
XXL	Prompt, Mix-Unsup	**37.4**	16.2	74.0
Base	Model	17.9	89.0	0.3
Base	Model, Mix-Unsup	**25.2**	**16.2**	**56.8**
XXL	Model	30.1	32.7	16.8
XXL	Model, Mix-Unsup	**32.4**	**17.0**	**32.4**

Mixing in unsupervised multilingual data generally helps prevent catastrophic forgetting. It significantly improves XGen capacities for ModelTuning. For PromptTuning, it provides a benefit where catastrophic forgetting is more severe.

Factorized prompts are helpful when Prompt Tuning shows the most severe forgetting

		Th		
Size	**Method**	SP-Rouge	LID_{EN}	LID_{TH}
Base	Prompt	17.3	34.3	33.5
Base	Prompt, Mix-Unsup	20.9	**4.1**	**76.9**
Base	Prompt, FP	**21.1**	19.8	40.0
XXL	Prompt	**37.4**	13.5	75.5
XXL	Prompt, Mix-Unsup	**37.4**	16.2	74.0
XXL	Prompt, FP	36.9	**9.0**	**80.8**

Factorized prompts are successful at improving target language accuracy. However, this does not always translate to higher SP-Rouge. In settings where vanilla PromptTuning shows the most severe forgetting (e.g., at Base size), factorized prompts provide large gains.

Take-aways

1. Prompt Tuning is preferred over Model Tuning when there is a significant language shift at inference time

2. Increasing model scale + decreasing tunable parameter capacity are both effective for xGen

3. Methods like mixing in unlabeled multilingual data and factorized prompts are helpful

R2: Can current transfer learning methods extend successfully to a zero-shot cross-lingual transfer setting?

➡ **significant headroom remains**

Future work: Parameter-efficient Multi-task Multimodal Multilingual Knowledge Sharing

How to share knowledge across tasks, modalities, and languages effectively and efficiently?

Thank you!

Q & A