Google AI

Massively Multilingual, Massive MT (M4)

Universal Translation at Scale

Aditya Siddhant, Ali Dabirmoghaddam, Ankur Bapna, Colin Cherry, Dmitry Lepikhin, George Foster, Isaac Caswell, James Kuczmarski, Kun Zhang, Macduff Hughes, Mahdis Mahdieh, Manisha Jain, Markus Freitag, Maxim Krikun, Melvin Johnson, Mia Chen, Naveen Arivazhagan, Orhan Firat, Roee Aharoni, Sébastien Jean, Sneha Kudugunta, Thang Luong, Wei Wang, Wolfgang Macherey, Yanping Huang, Yonghui Wu, Yuan Cao, Zhifeng Chen

Dec 16, 2019

Microsoft - Redmond

Agenda

1. **Goal & Motivations**
2. Project Phases
3. Open Problems

Our goal

Develop a universal machine translation model (i.e. one model for all languages and domains)

*"Perhaps the way [of translation] is to descend, from each language, down to the common base of human communication – the real but as yet **undiscovered universal language** – and then re-emerge by whatever particular route is convenient."*

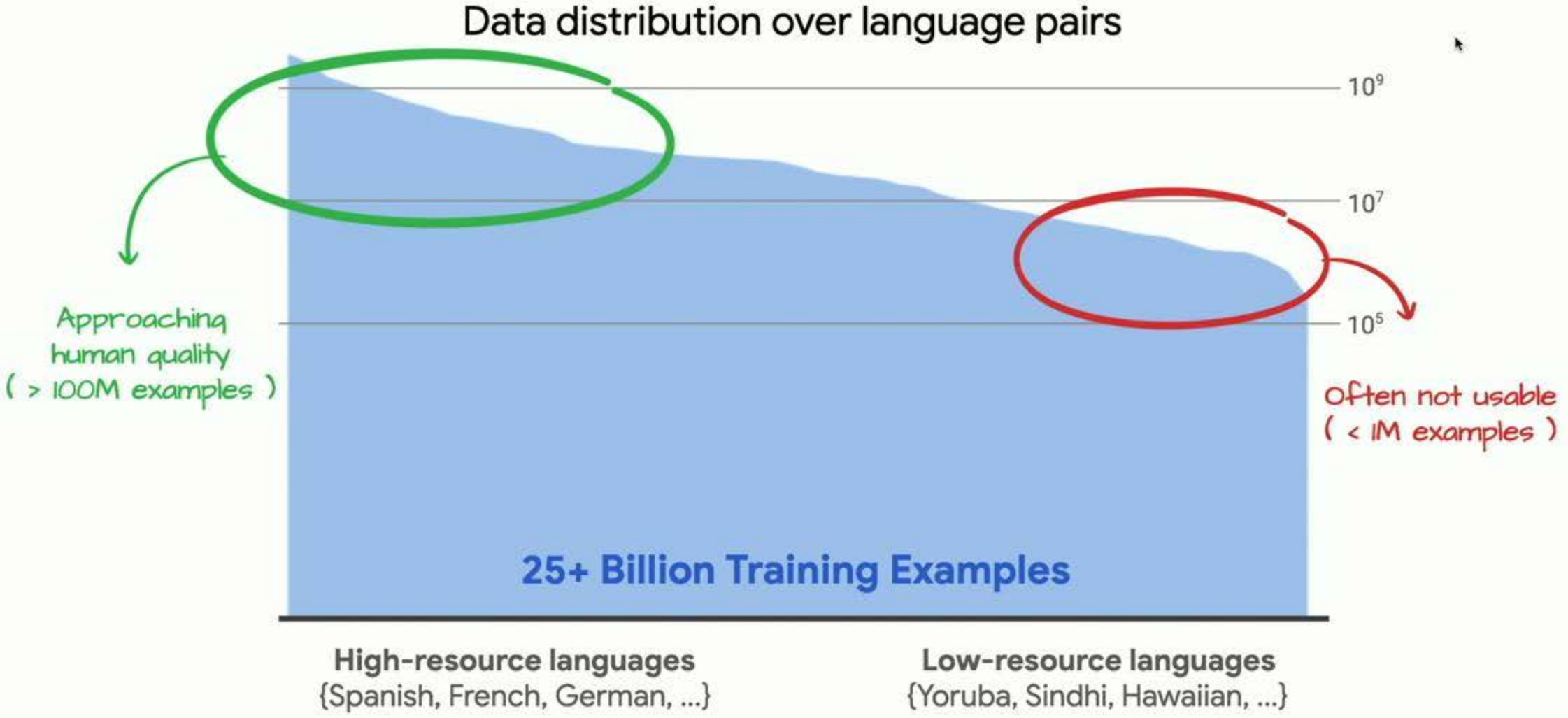
Warren Weaver (1949)

Google AI Blog

The latest news from Google AI

Exploring Massively Multilingual, Massive Neural Machine Translation

Motivation 1:
Improve translation quality for all language pairs

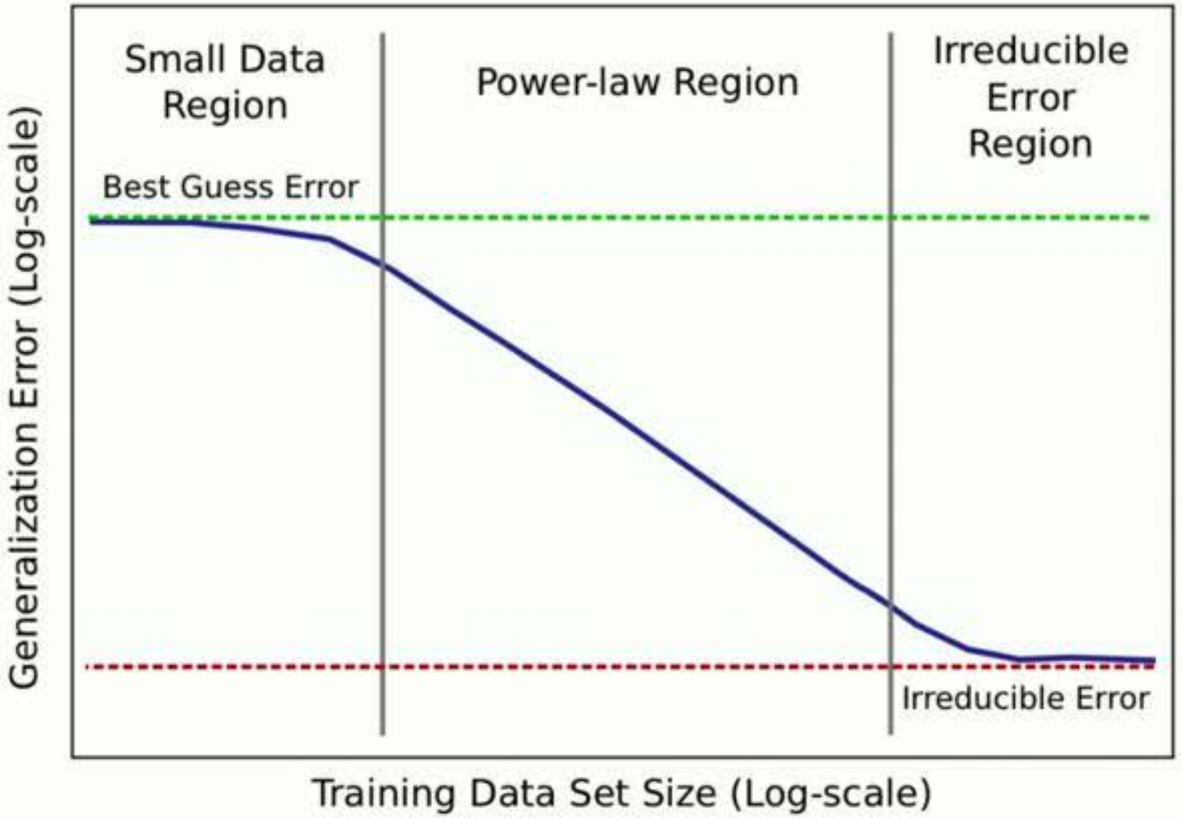


Motivation 2:
Expand language coverage

In the world, there are...	7,000+ Total languages	2,000+ African languages	700+ Native Am. languages[1]
But Translate only supports...	103 Total languages	11 African languages	0 Native Am. languages

1.. Estimate is 766 Native Am languages globally: 383 in South America (source); 176 in central America (source); 207 in North America (source)

Motivation 3:
Neural network scaling and the new understanding of generalization



Hestness et al. 2018

Motivation 4:
This is a compelling test bed for ML research

Massive multilinguality requires advances in :

- Multi-task learning
- Meta-learning
- Continual learning

To achieve massive multilinguality, we need massive scale, requires advances in:

- Model capacity
- Trainability and optimization
- Efficiency improvements

Agenda

1. Goal & Motivation
2. **Project Phases**
3. Open Problems

To achieve our goal, we knew we had to dramatically scale capacity; but first, we had to enumerate all relevant research challenges

Conference Publications

- EMNLP
- NAACL
- NeurIPS

Training Deeper Neural Machine Translation Models with Transparent Attention

Ankur Bapna * **Mia Xu Chen** * **Orhan Firat** * **Yuan Cao** * **Yonghui Wu**
ankurbpn,miachen,orhanf,yuancao@google.com
Google AI

Massively Multilingual Neural Machine Translation

Roee Aharoni* Bar Ilan University Ramat-Gan Israel roee.aharoni@gmail.com

Melvin Johnson and **Orhan Firat** Google AI Mountain View California melvinp,orhanf@google.com

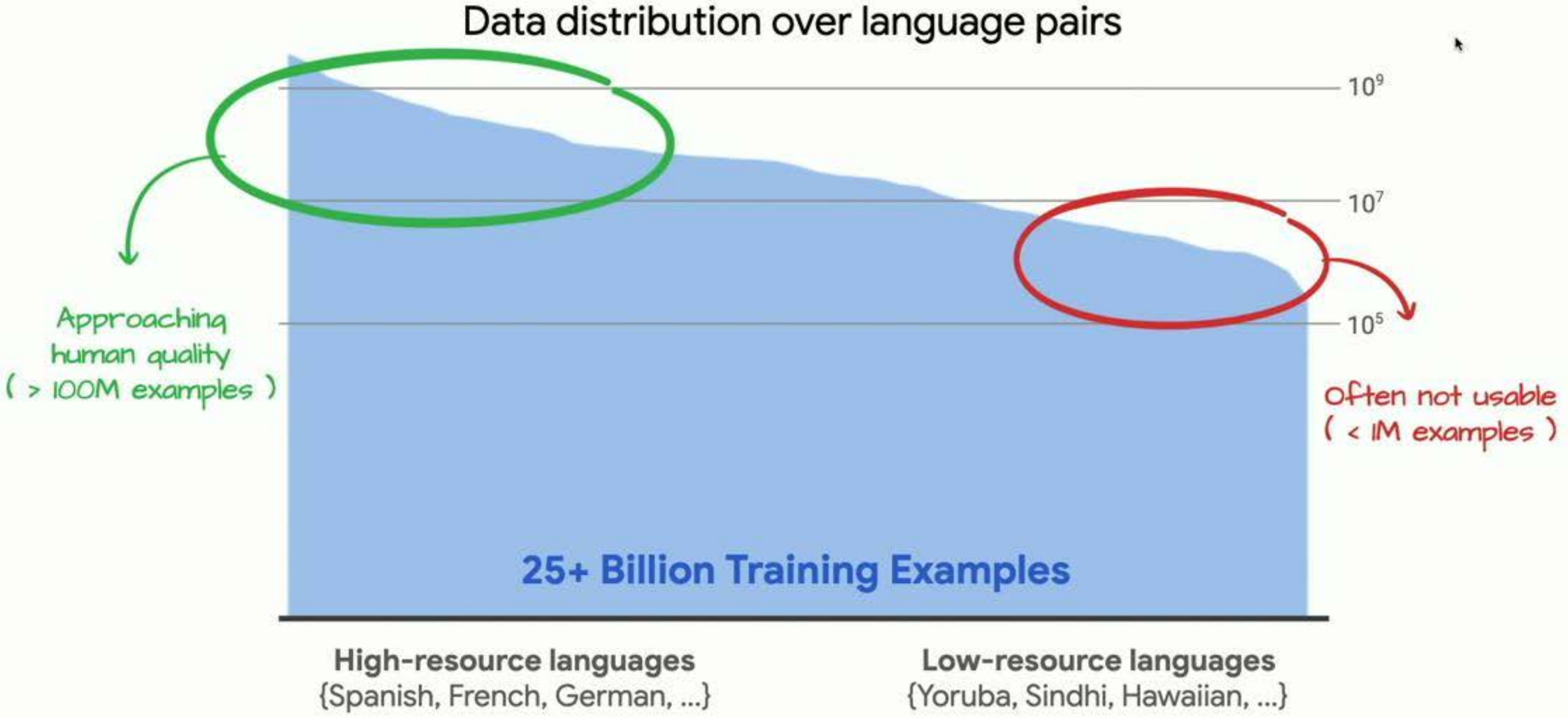
Adaptive Scheduling for Multi-Task Learning

Sébastien Jean* Department of Computer Science New York University sebastien@cs.nyu.edu

Orhan Firat Google AI orhanf@google.com

Melvin Johnson Google AI melvinp@google.com

Motivation 1:
Improve translation quality for all language pairs



To achieve our goal, we knew we had to dramatically scale capacity; but first, we had to enumerate all relevant research challenges

Conference Publications

- EMNLP
- NAACL
- NeurIPS

Training Deeper Neural Machine Translation Models with Transparent Attention

Ankur Bapna * **Mia Xu Chen** * **Orhan Firat** * **Yuan Cao** * **Yonghui Wu**
ankurbpn,miachen,orhanf,yuancao@google.com
Google AI

Massively Multilingual Neural Machine Translation

Roee Aharoni* Bar Ilan University Ramat-Gan Israel roee.aharoni@gmail.com

Melvin Johnson and **Orhan Firat** Google AI Mountain View California melvinp,orhanf@google.com

Adaptive Scheduling for Multi-Task Learning

Sébastien Jean* Department of Computer Science New York University sebastien@cs.nyu.edu

Orhan Firat Google AI orhanf@google.com

Melvin Johnson Google AI melvinp@google.com

After our pilot studies, we moved to realistic scenario: M4 in the wild

RESEARCH PRIORITIES

- Develop baselines
- Learn, given data imbalance
- Increase model capacity

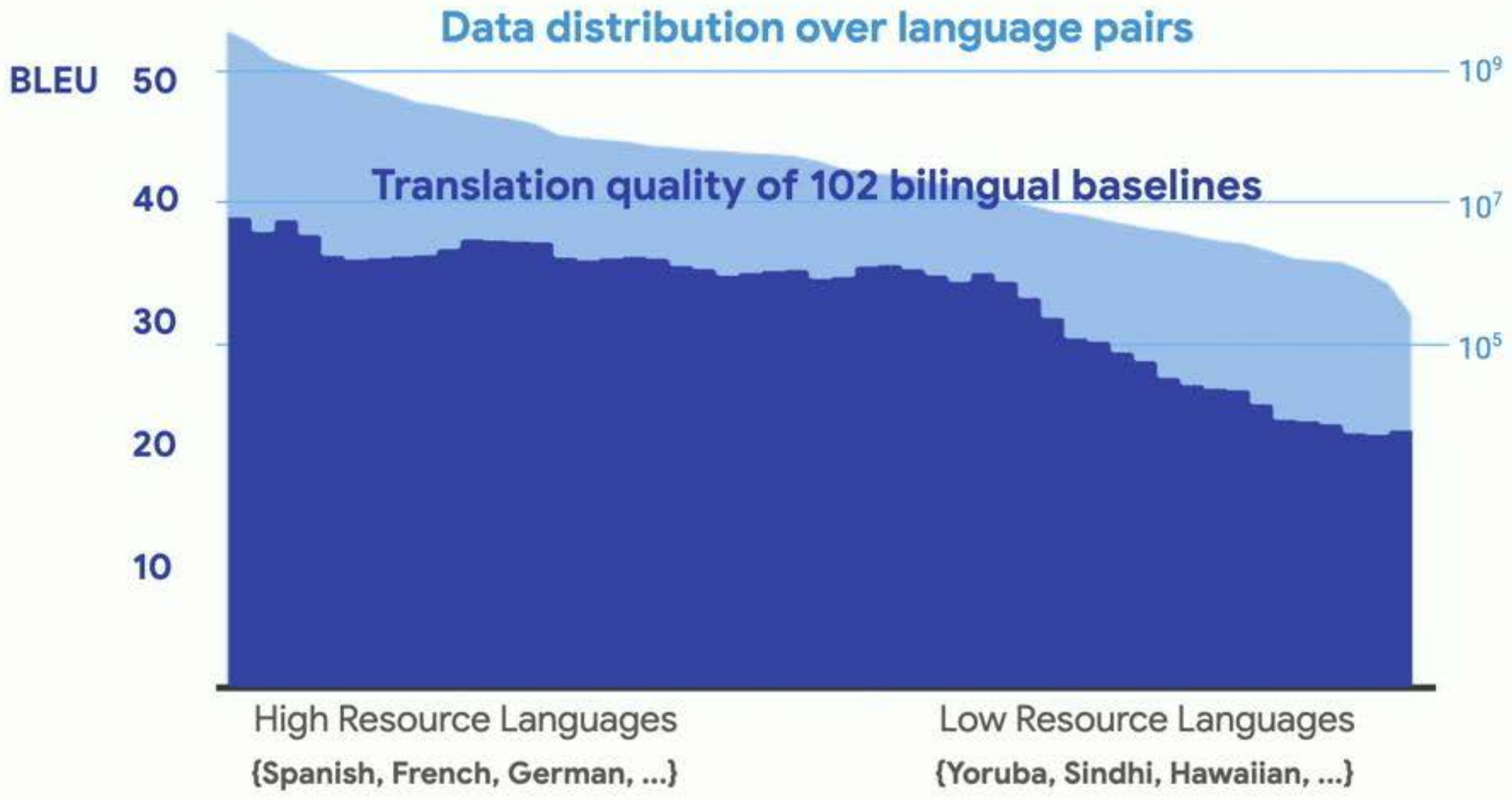
Let's go into detail on each priority

GOAL

Train a 103-language model; attain parity with baselines

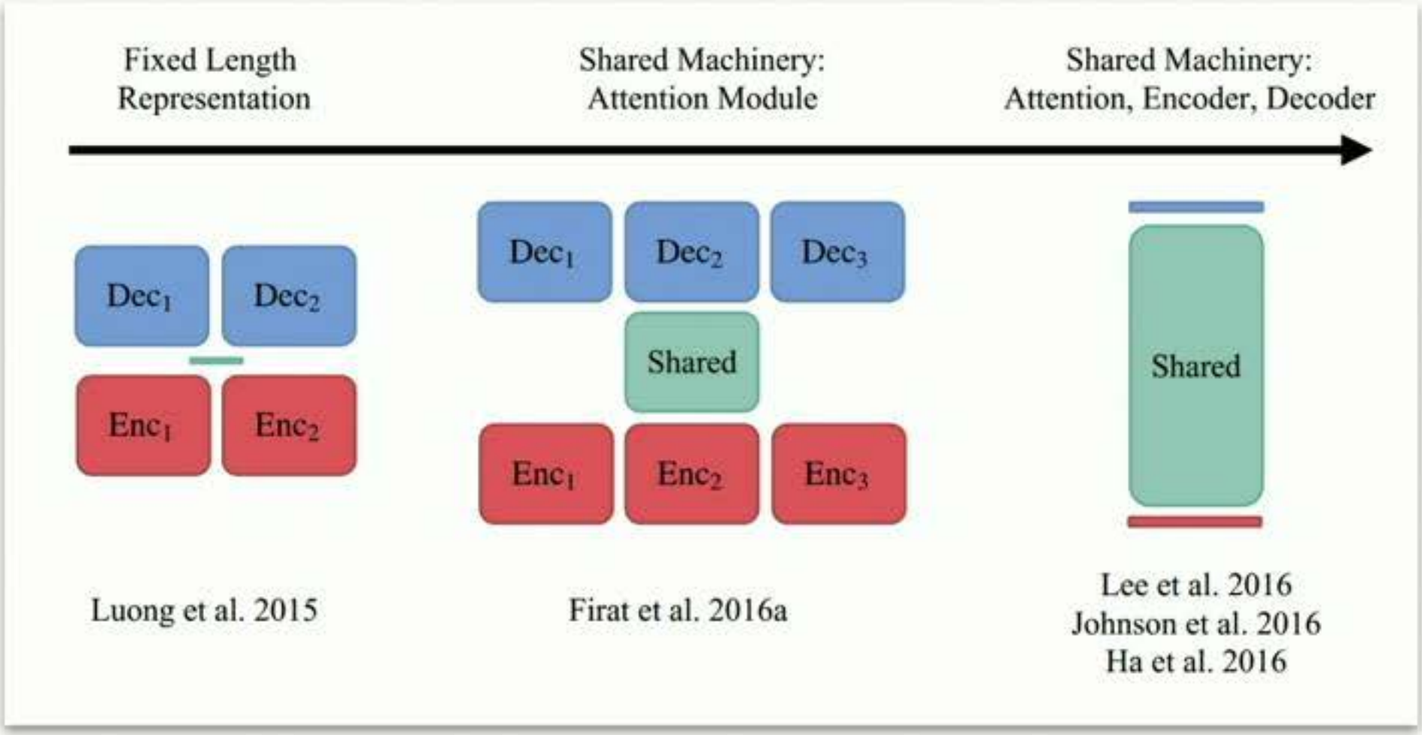
1/ Develop baselines:

We trained and evaluated new bilingual models as controls



1/ Develop baselines:

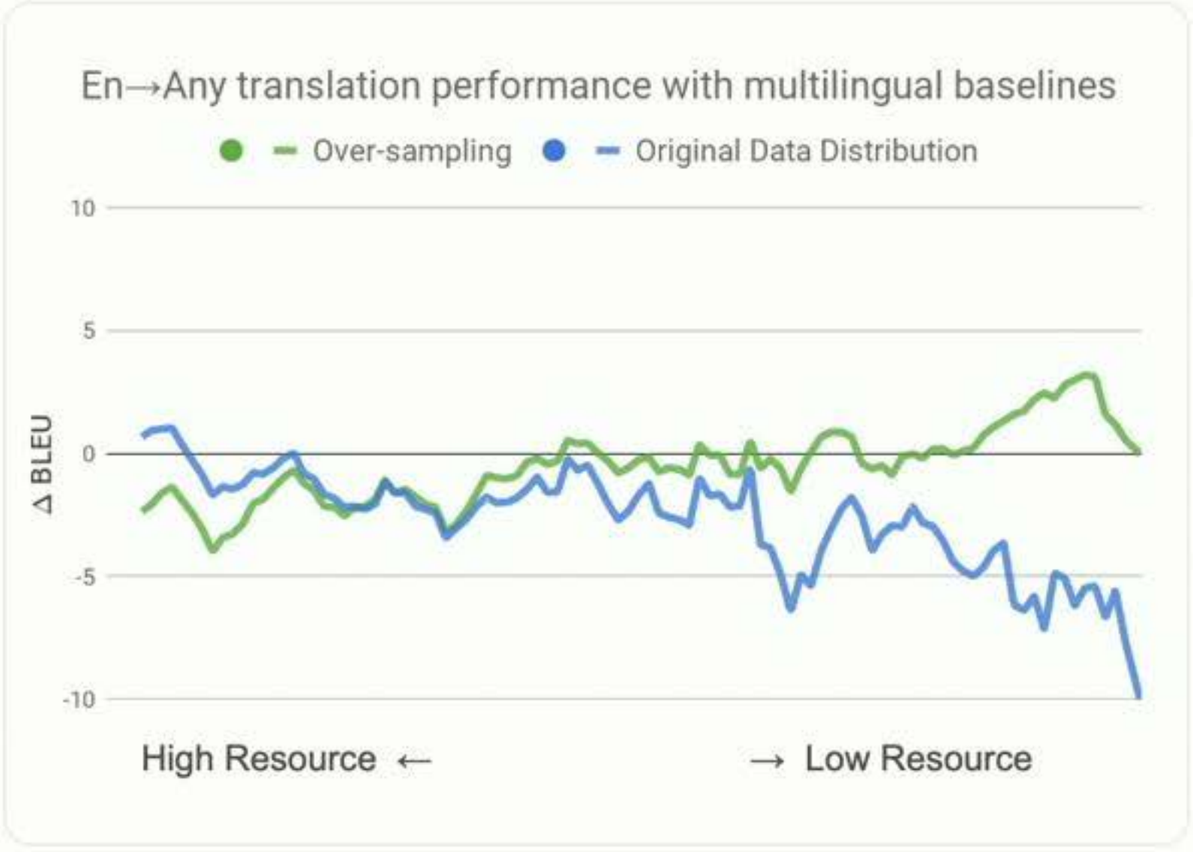
Models
Wiring: Transformer as explained in [1] and [2]



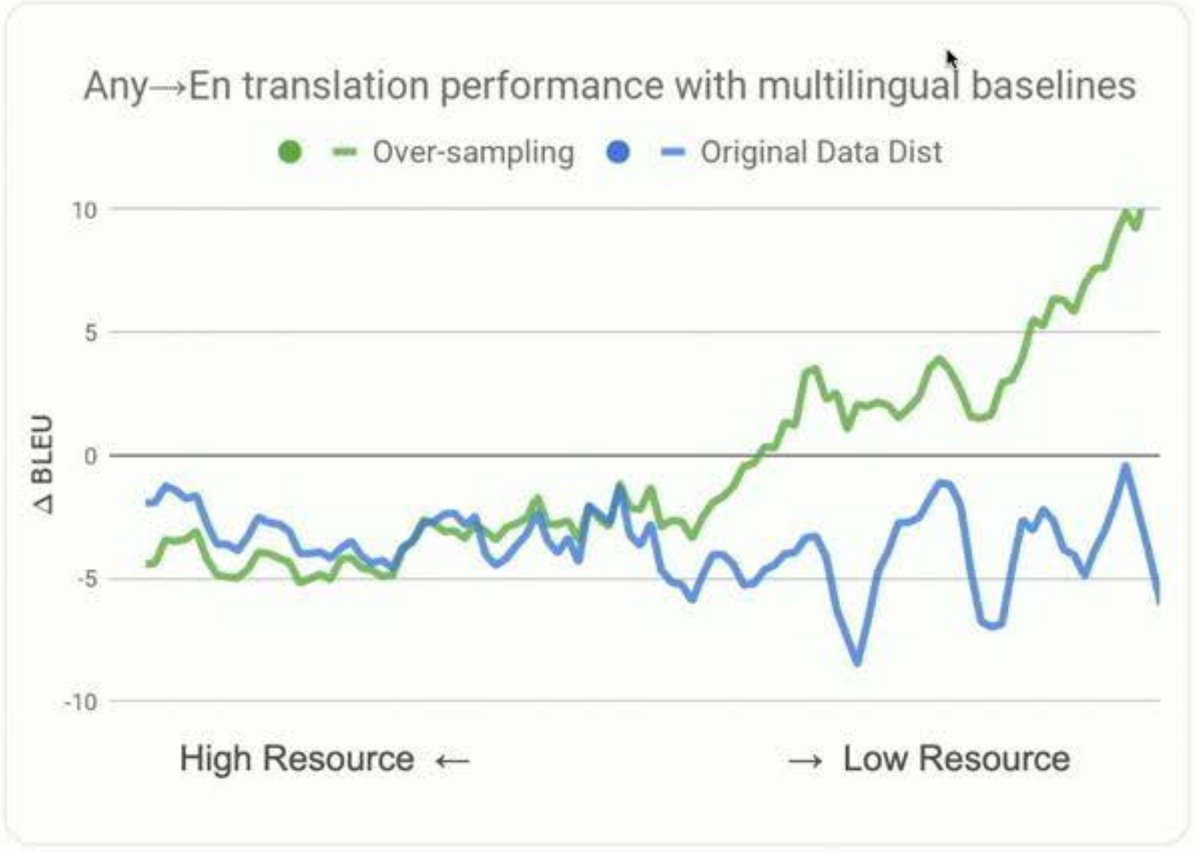
[1] Chen et al. 2018, "The Best of Both Worlds: Combining Recent Advances in Neural Machine Translation"
[2] Bapna et al. 2018, "Training Deeper Neural Machine Translation Models with Transparent Attention"

2/ Learn, given data imbalance:

Importance of re-balancing data



English to Any

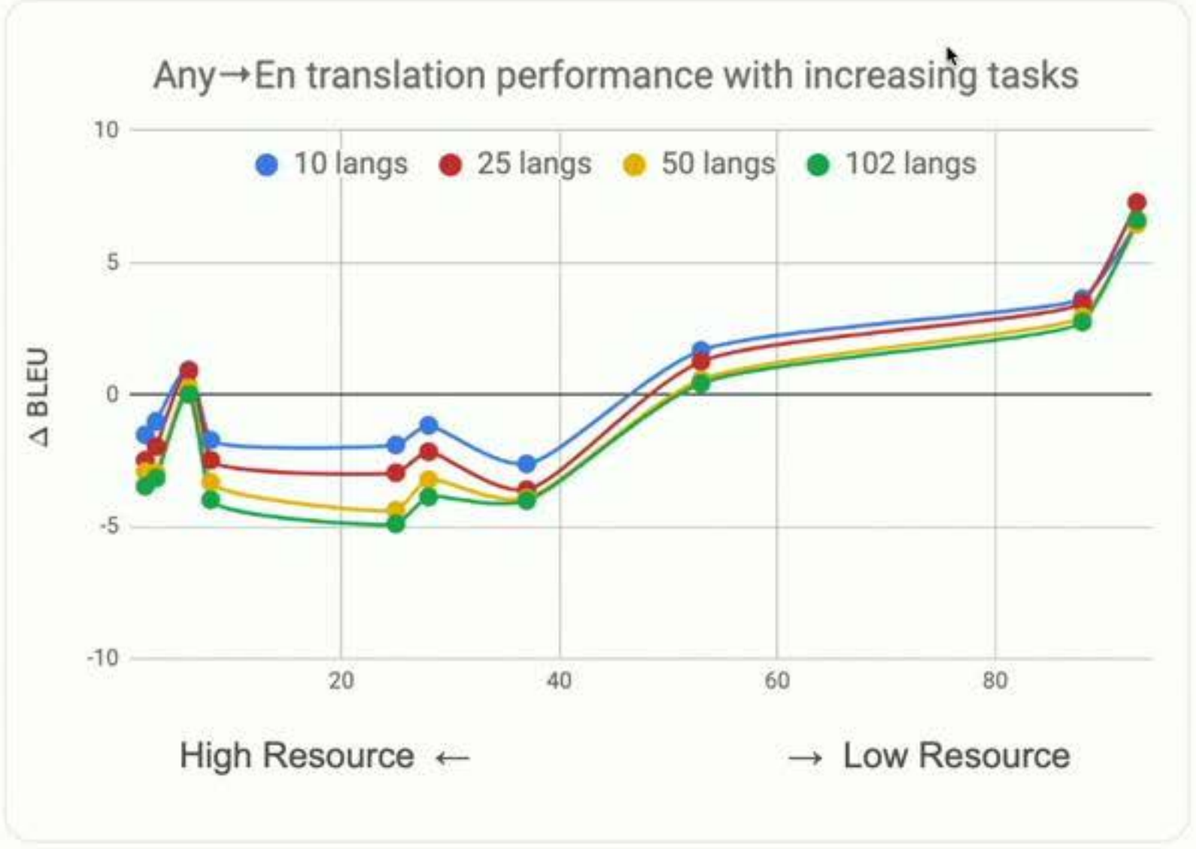


Any to English

2/ Learn, given data imbalance:

Increasing the number of languages at the same capacity results in worse quality due to interference, especially in En→Any

English to Any



Any to English

2/ Learn, given data imbalance:

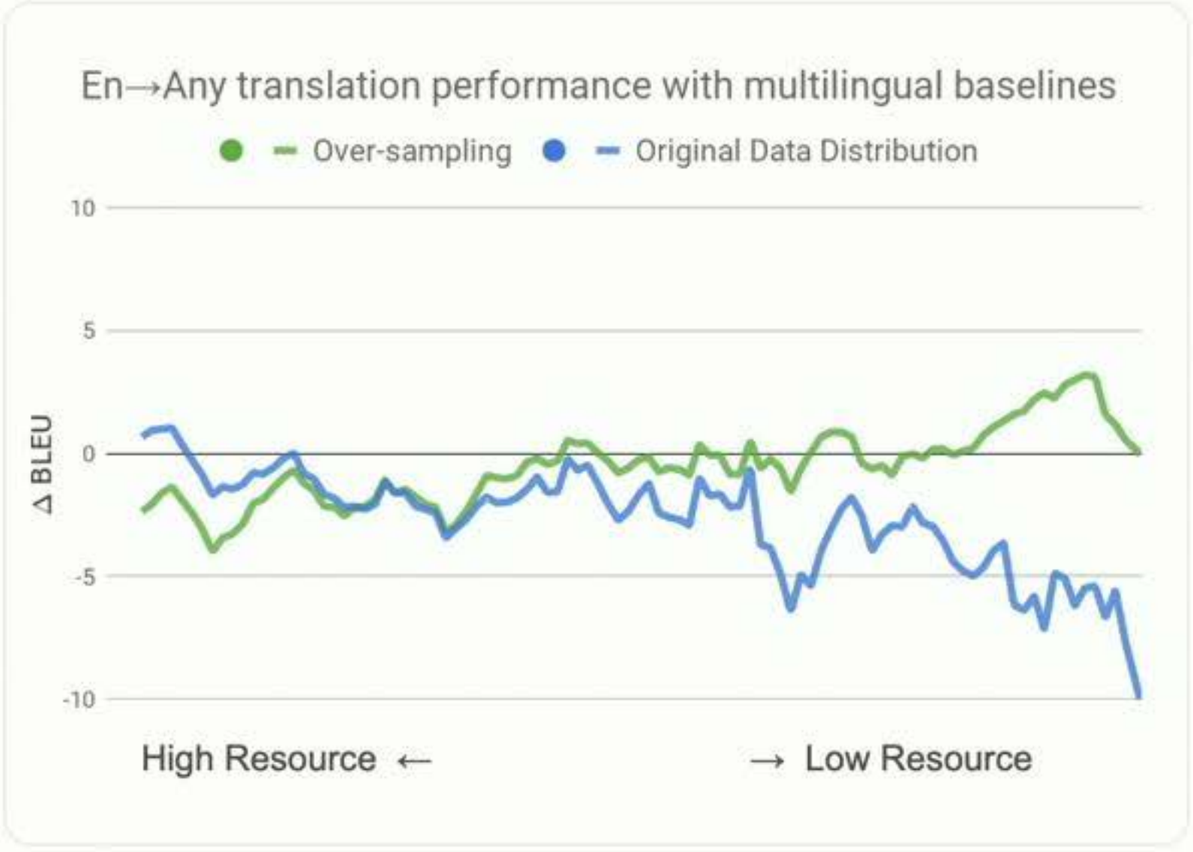
Two separate models: En→Any and Any→En

English to Any

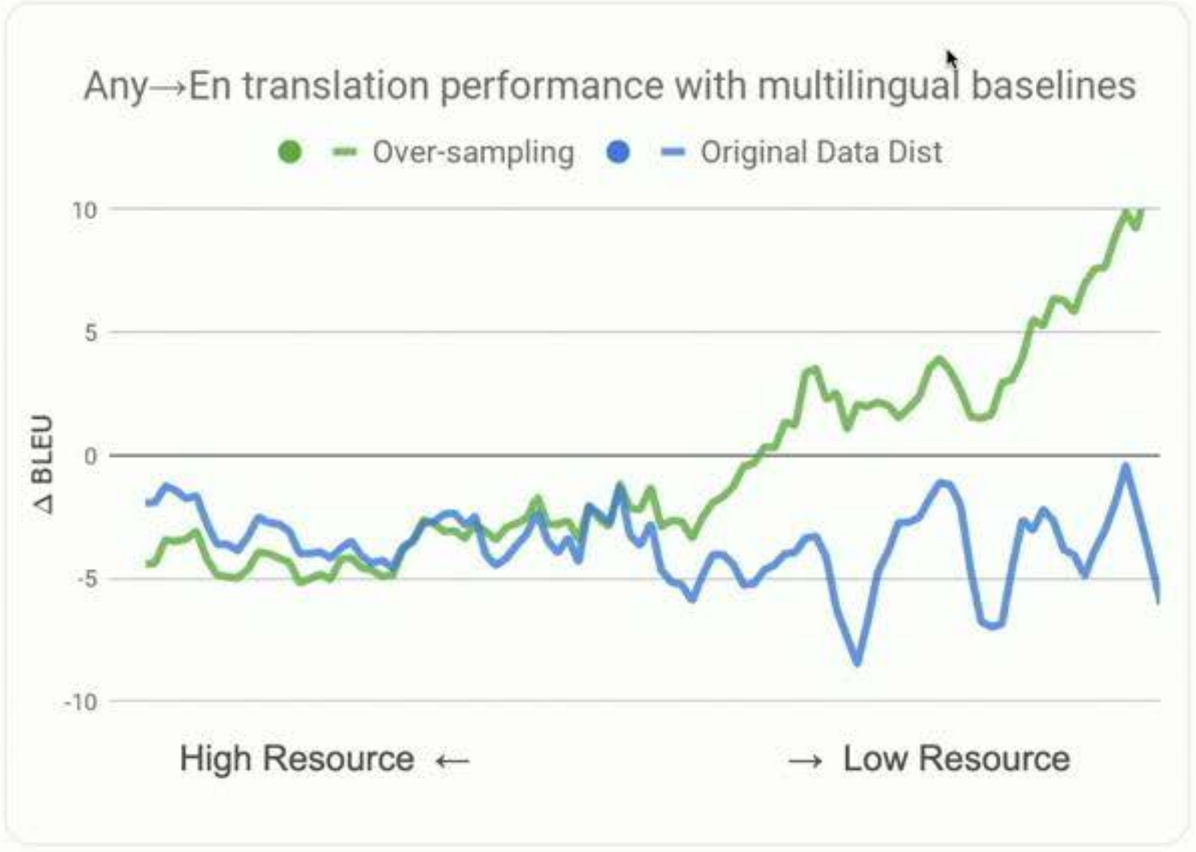
Any to English

2/ Learn, given data imbalance:

Importance of re-balancing data



English to Any

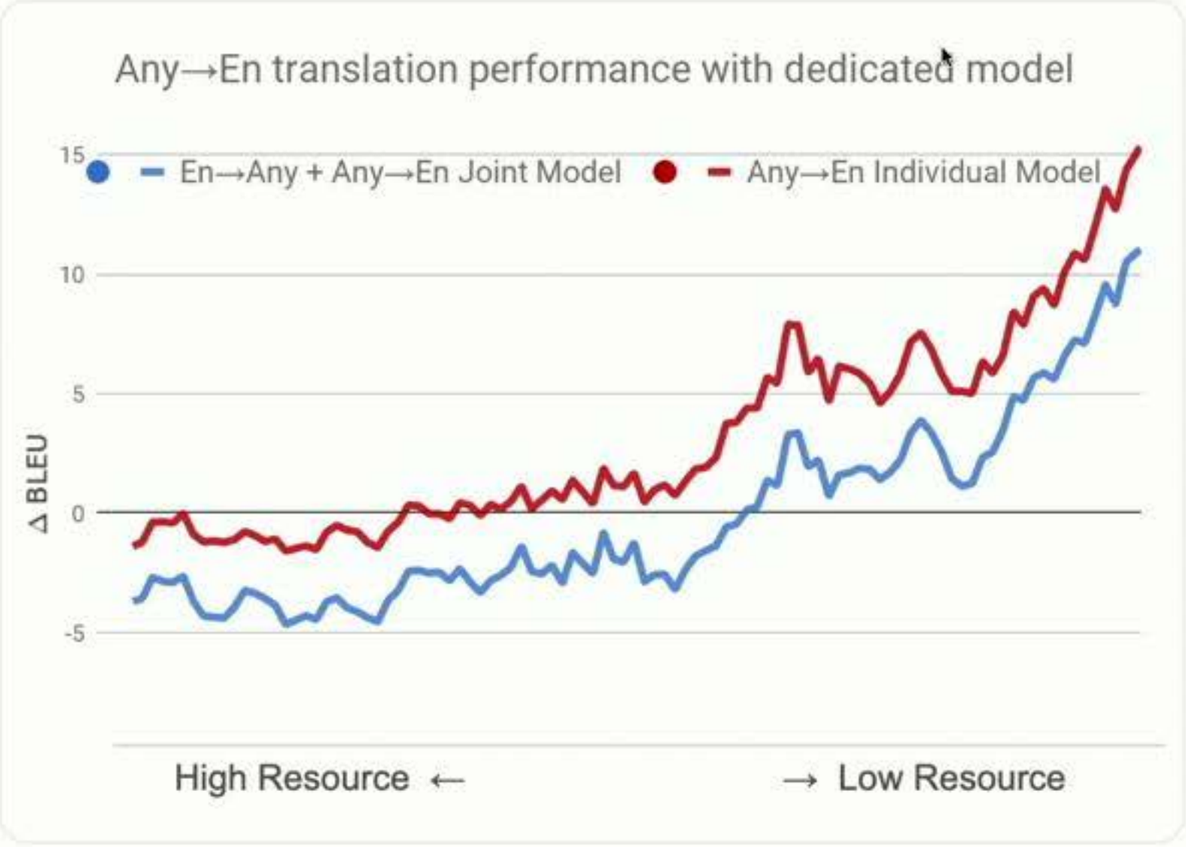


Any to English

2/ Learn, given data imbalance:

Two separate models: En→Any and Any→En

English to Any



Any to English

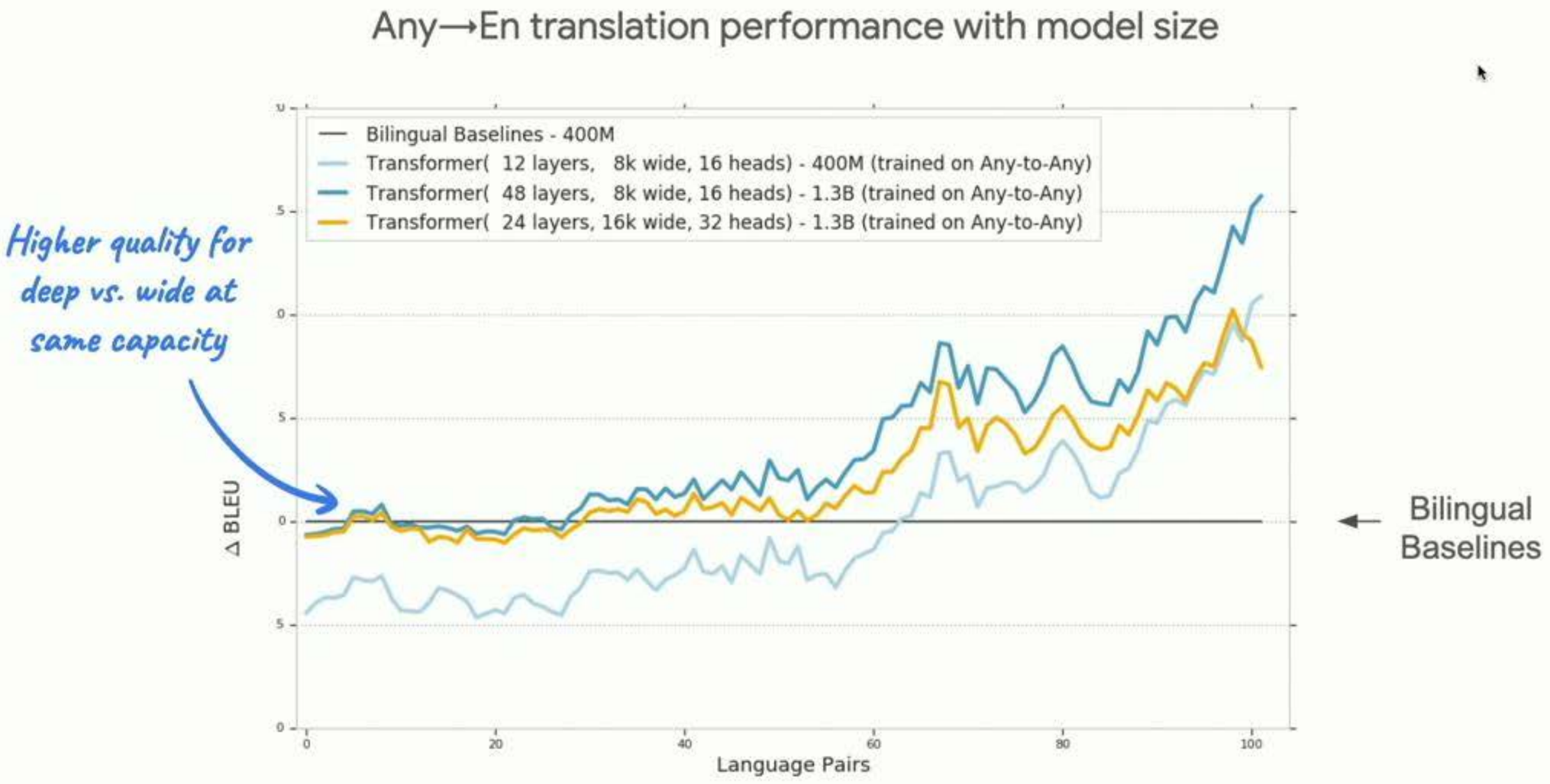
To reduce quality losses, we needed greater model capacity

Relevant research questions:

- Do deep or wide networks drive greater quality gains?
- How deep or wide should we go?
- What quality boost do we attain by sacrificing En→Any (i.e. half of our tasks)?

3/ Increase model capacity:

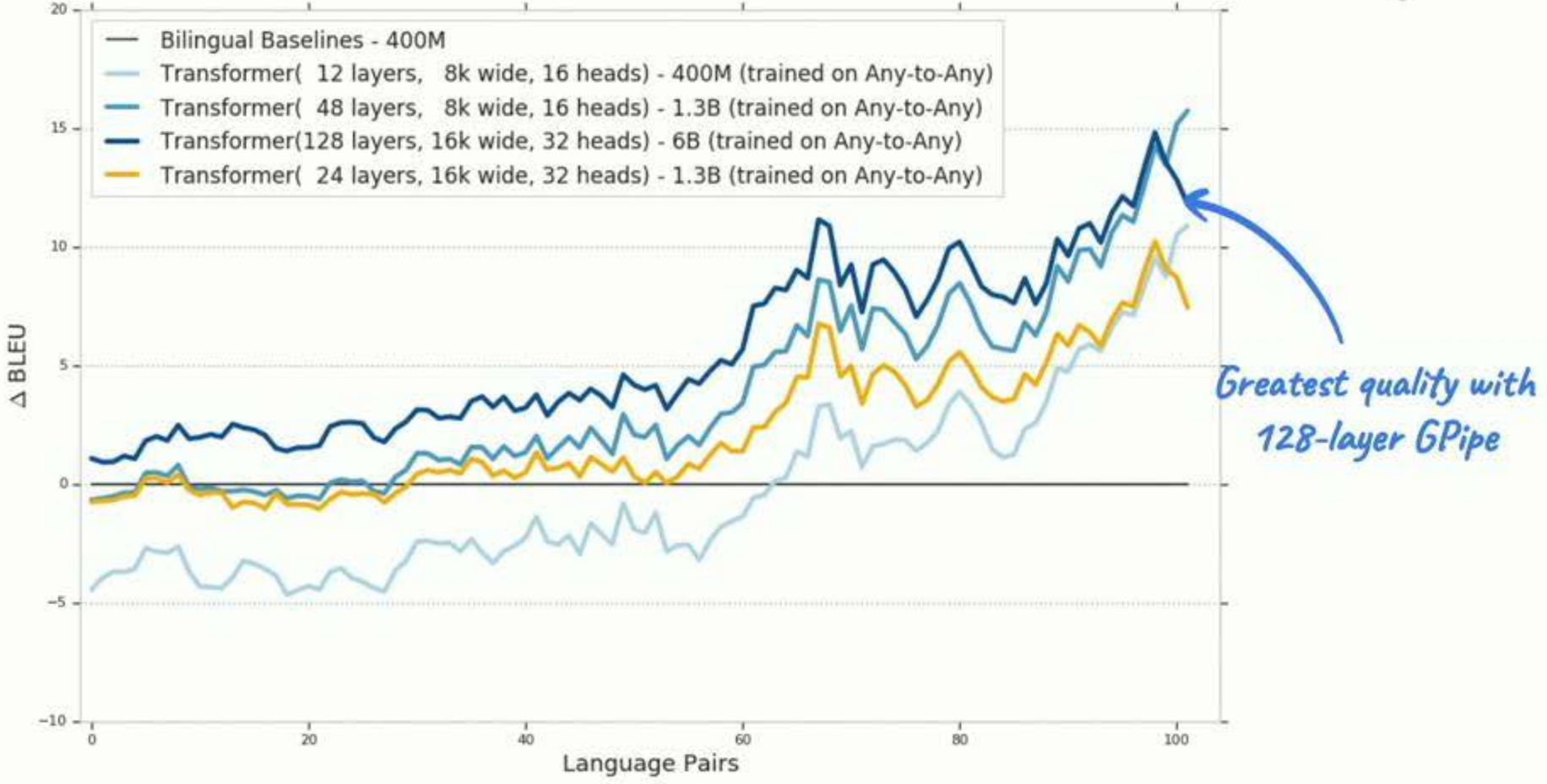
Do deep or wide networks drive greater quality gains?



3/ Increase model capacity:

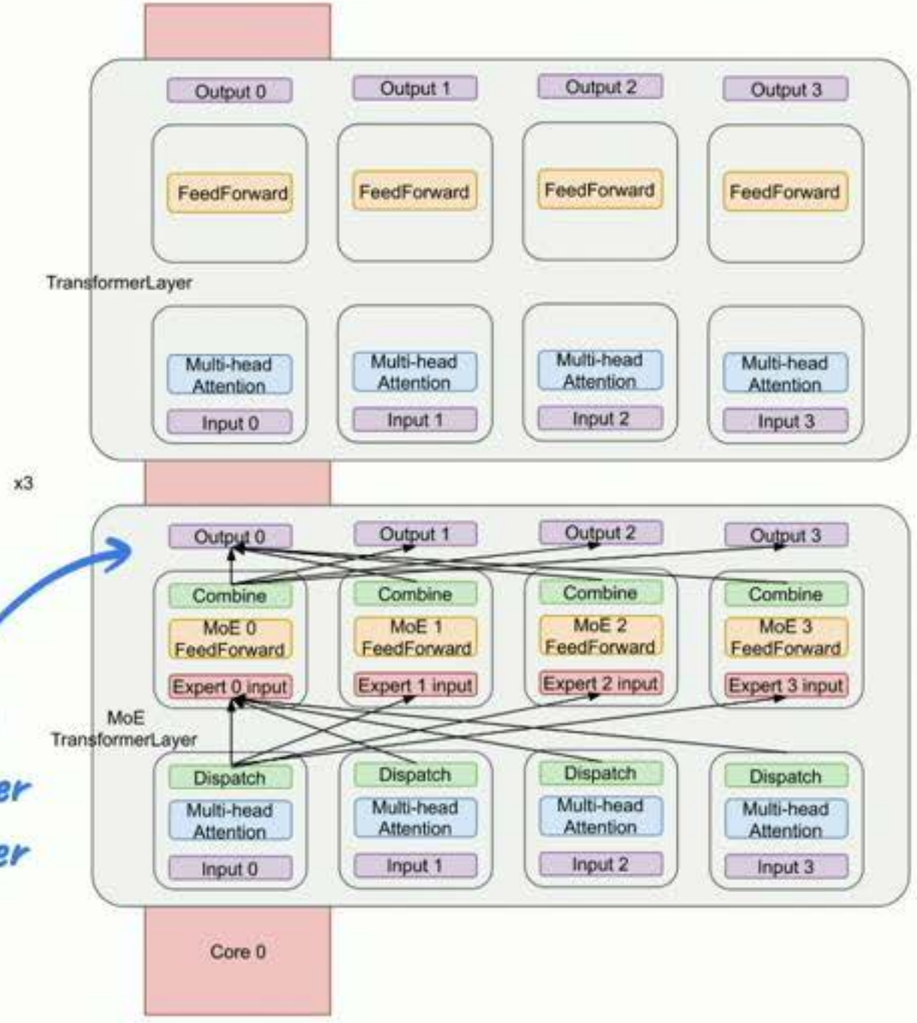
How deep or wide should we go?

All-to-English Translation Quality for M4 Models.



3/ Increase model capacity:

How can we scale to 1k chips and beyond?
Conditional computation and Mixture-of-Experts with 50B+ weights



Replaces every other Transformer feed-forward layer with MoE

As we increase number of experts, training and inference scales sublinearly:

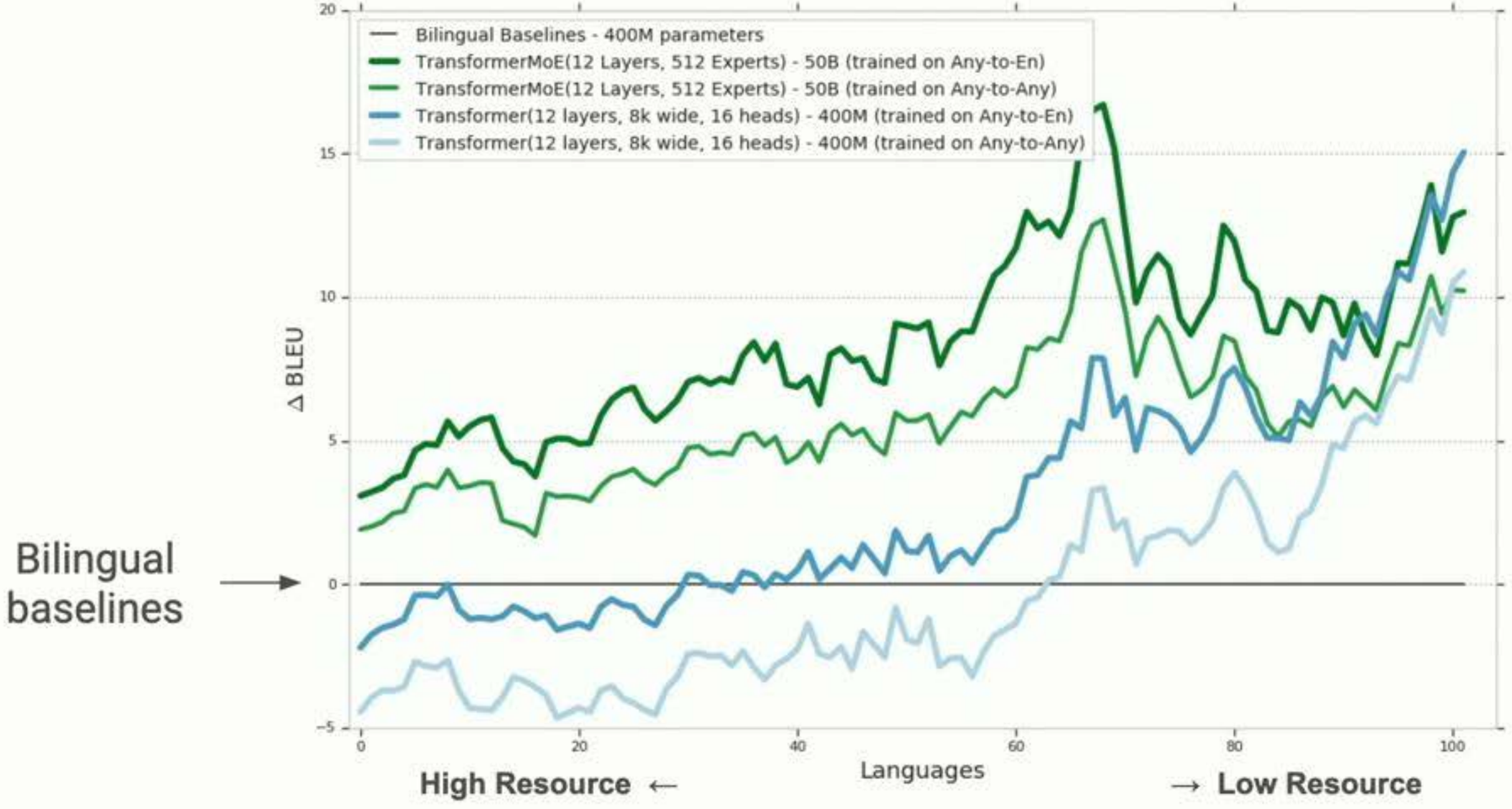
- FLOPS ∝ batch_size * avg. gated subnetwork size

Tradeoffs of 512 token-level experts & 1 expert/core:

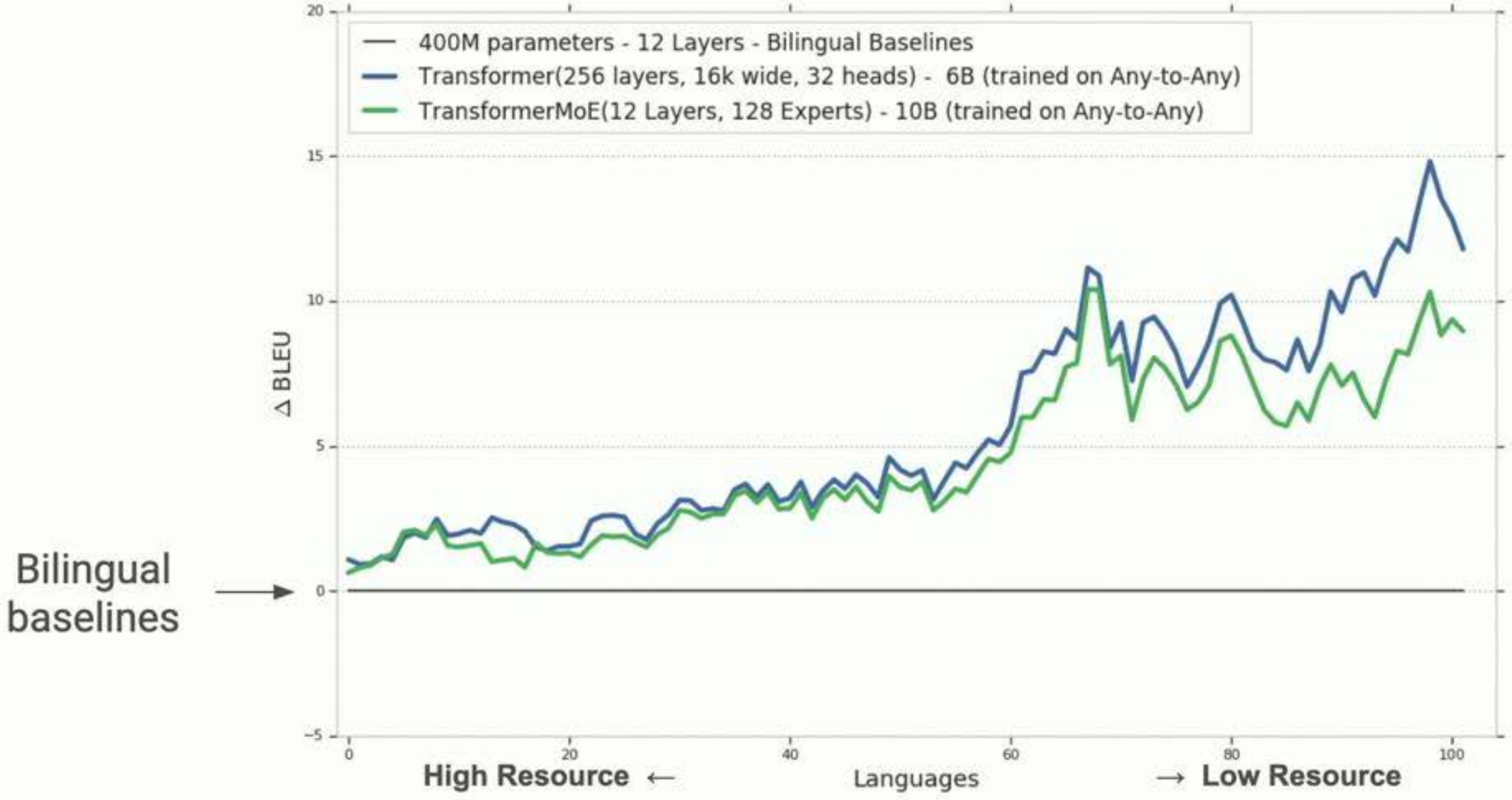
- Don't need MoE gradient aggregation
- MoE weight update broadcast is not required
- However, an extra network is required to dispatch activations

3/ Increase model capacity:

What quality boost do we attain by sacrificing En→Any (i.e. half of our tasks)?



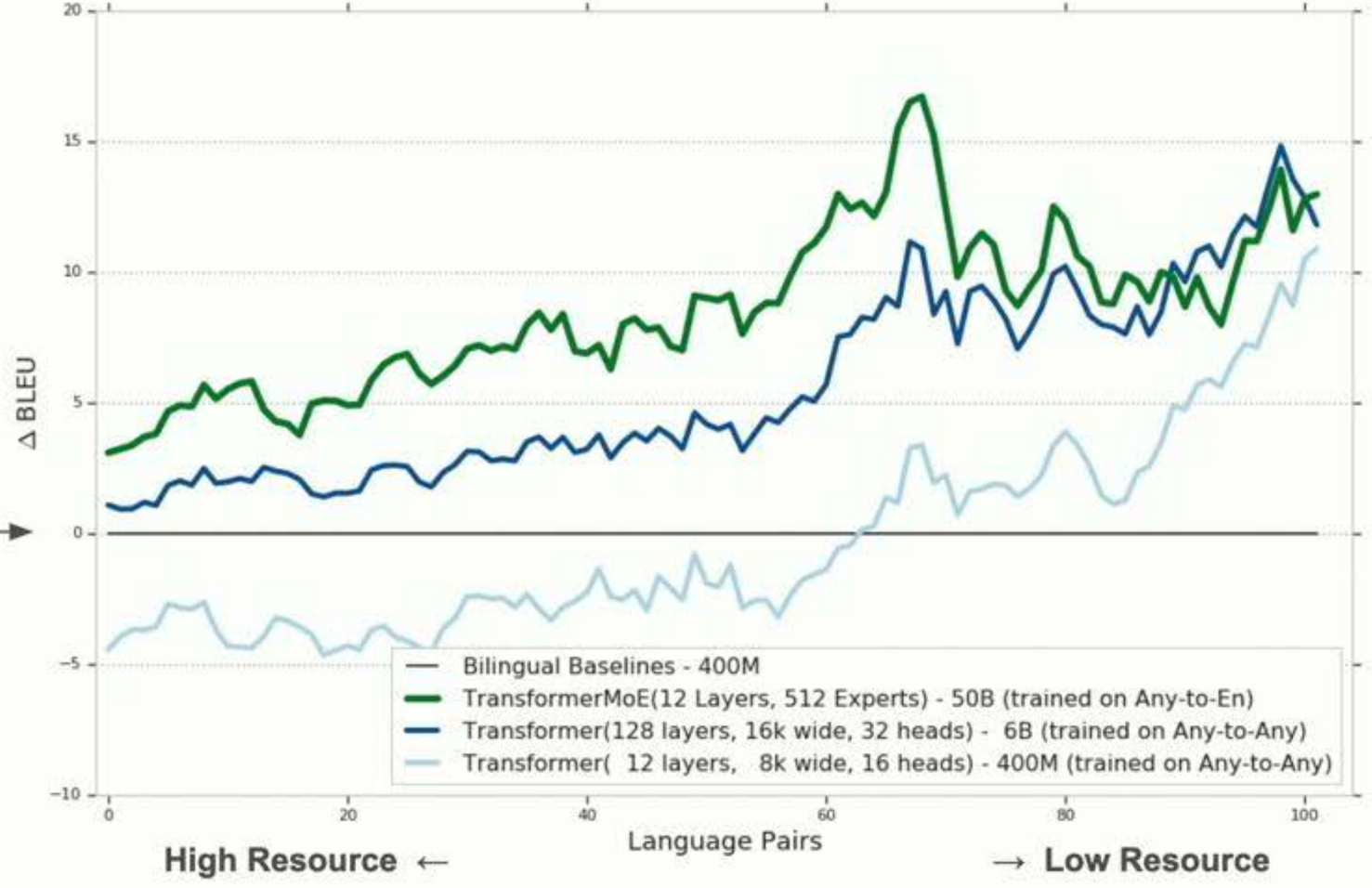
Parameter Efficiency: At similar parameter count, **deep models** perform better than **MoE**, but what about Flops?



3/ Increase model capacity:

Final Phase 1 results

Bilingual baselines →



Number of Synapses

#synapses
[wiki]

10^{6}> 10^{9} 10^{12} 10^{13} 10^{14} 10^{15}

Fruit fly

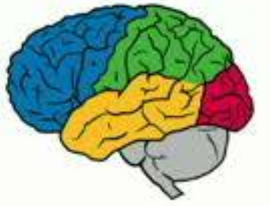
Honey bee

Mouse

Cat

Macaque

Human



Number of Synapses

NMT with Attention
Resnet50
[25-50M]

Fruit fly

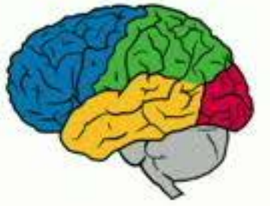
Honey bee

Mouse

Cat

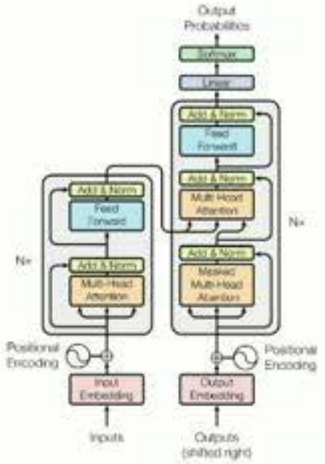
Macaque

Human



Google AI

Number of Synapses



Transformer [400M]

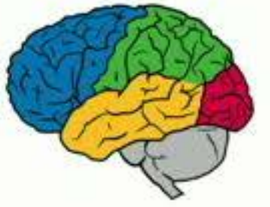
#synapses [wiki]

10^{6}> 10^{9} 10^{12} 10^{13} 10^{14} 10^{15}

Fruit fly **Honey bee** **Mouse** **Cat** **Macaque** **Human**

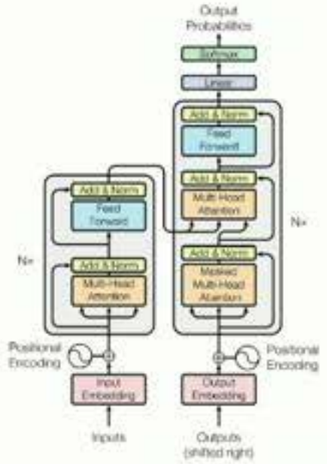






Google AI

Number of Synapses



Transformer
[400M]

Facebook ResNeXt101
Open AI GPT2
MSR ZeRO
NVidia Megatron-LM
Google-T5
[1-10B]

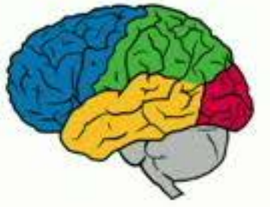
10^{6}> 10^{9} 10^{12} 10^{13} 10^{14} 10^{15}

#synapses
[wiki]

Fruit fly **Honey bee** **Mouse** **Cat** **Macaque** **Human**

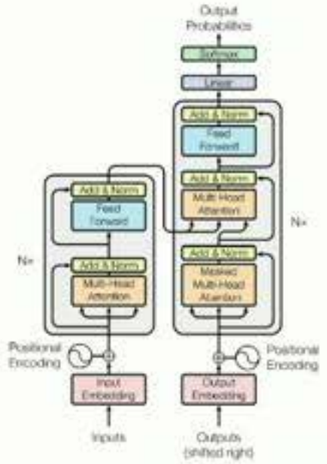






Google AI

Number of Synapses



Transformer
[400M]

Facebook ResNeXt101
Open AI GPT2
MSR ZeRO
NVidia Megatron-LM
Google-T5
[1-10B]

M4:
Massively Multilingual Massive Machine Translation
[80B]

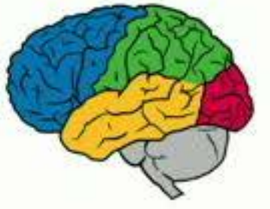
10^{6}> 10^{9} 10^{12} 10^{13} 10^{14} 10^{15}

#synapses [wiki]

Fruit fly **Honey bee** **Mouse** **Cat** **Macaque** **Human**

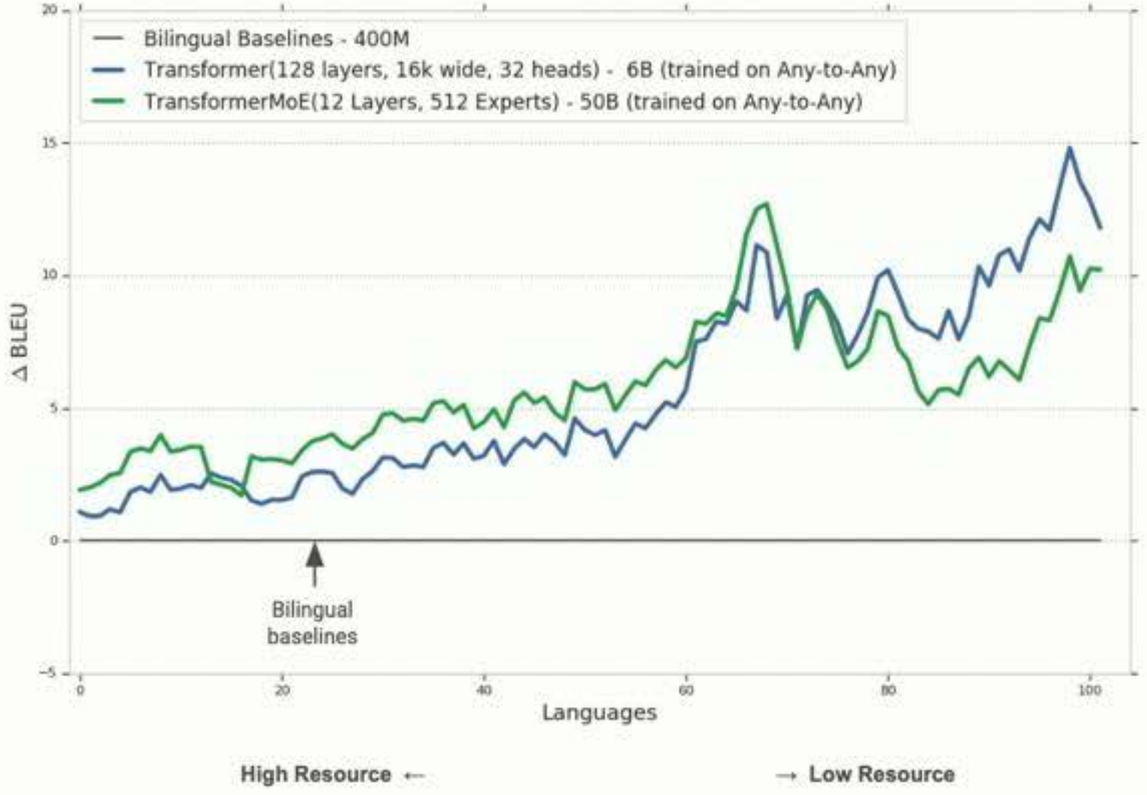




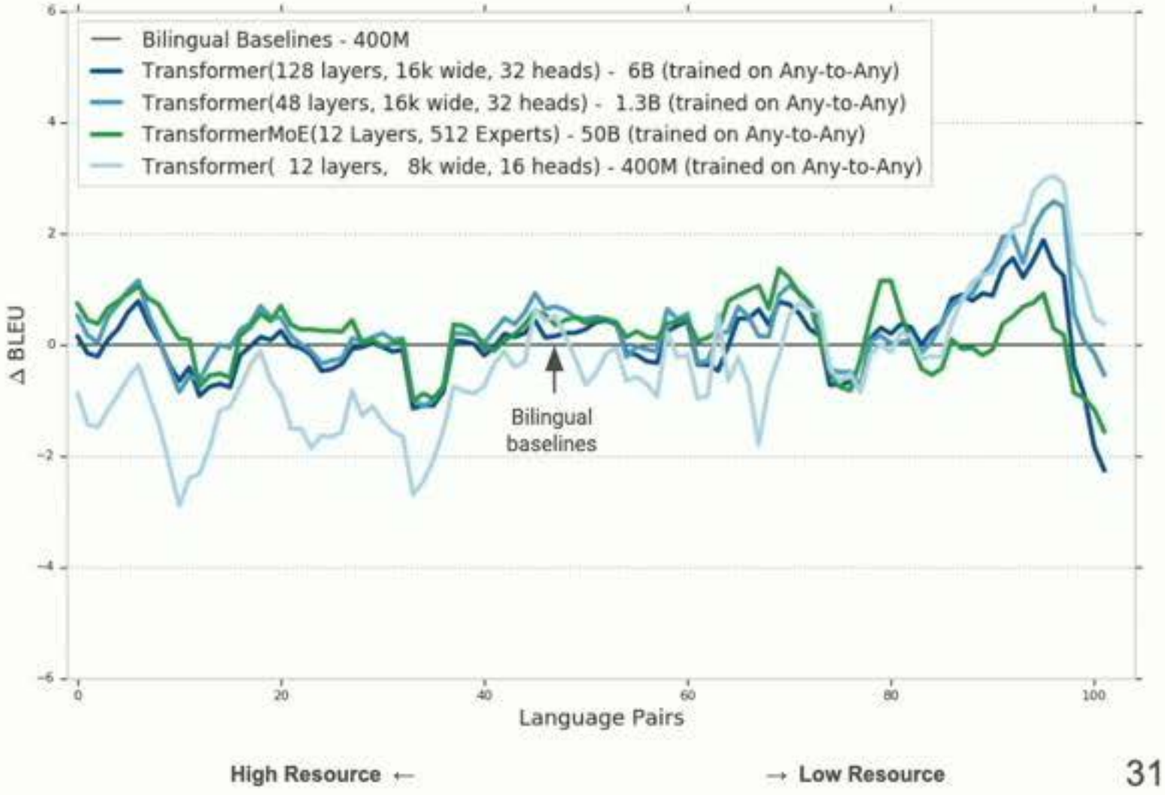


At the end of the first phase, two other insights helped inform research questions for the next phases

1/ We need a future model that can get the "best of both worlds" across MoE and Deep Transformer



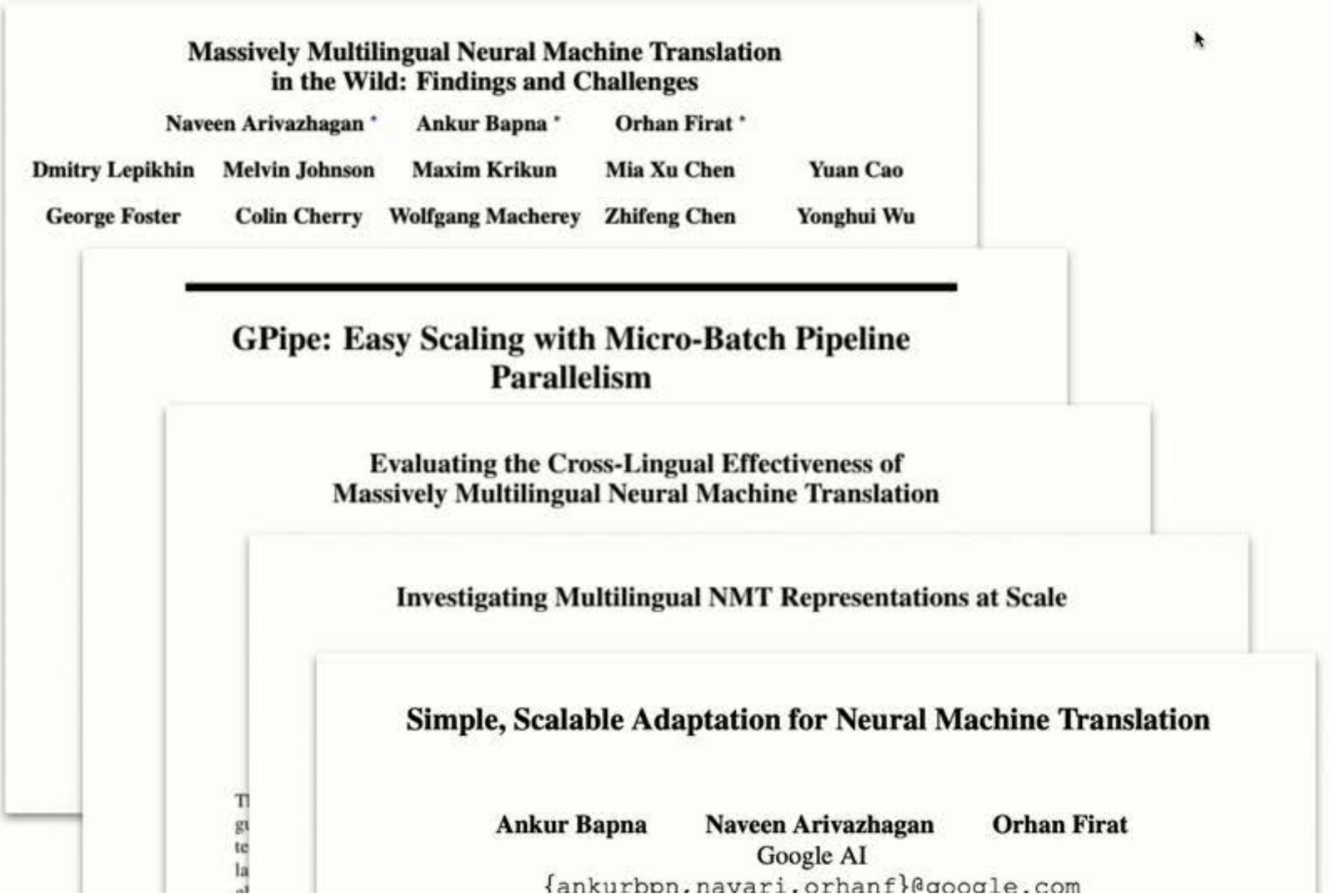
2/ Scaling up model capacity doesn't immediately improve performance on En→Any



Summary of recent publications

Conference Publications

- NeurIPS
- EMNLP
- AAAI



Massive Neural Networks

Training Deeper Neural Machine Translation Models with Transparent Attention, Bapna et al. EMNLP 2018

GPipe: Efficient Training of Giant Neural Networks using Pipeline Parallelism, Huang et al. NeurIPS 2019

Outrageously Large Neural Networks: The Sparsely-Gated Mixture-of-Experts Layer, Shazeer et al. ICLR 2017

GPipe: Easy Scaling with Pipeline Parallelism – Huang et al., 2019

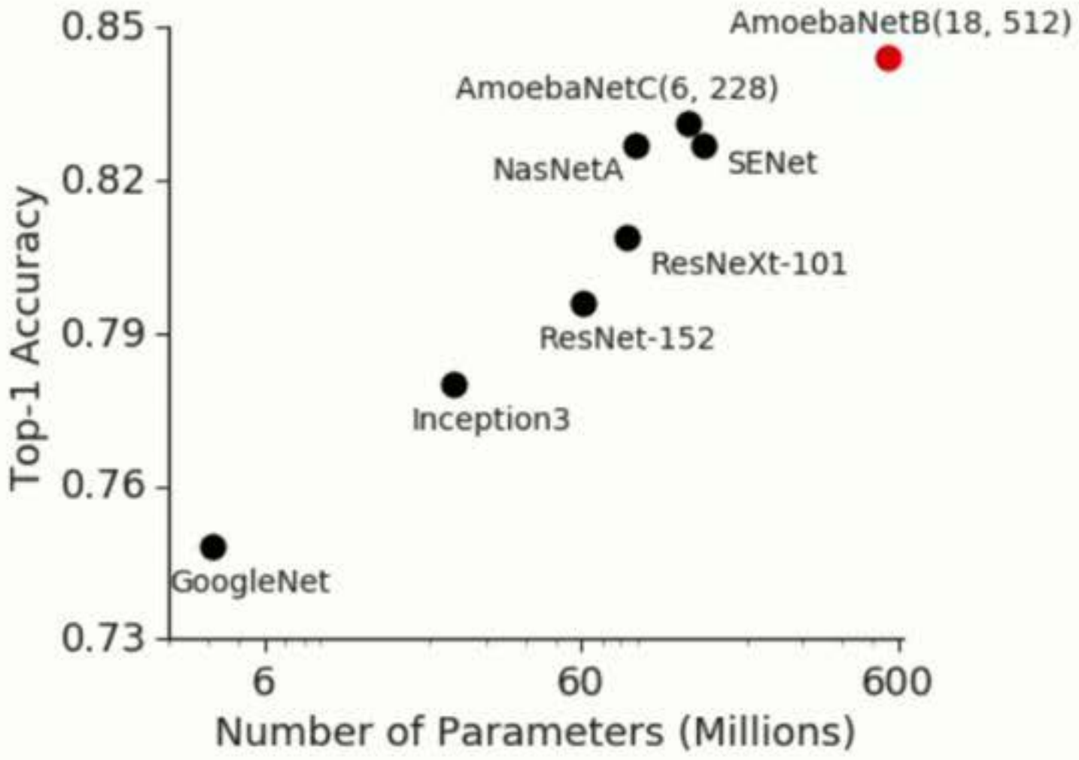
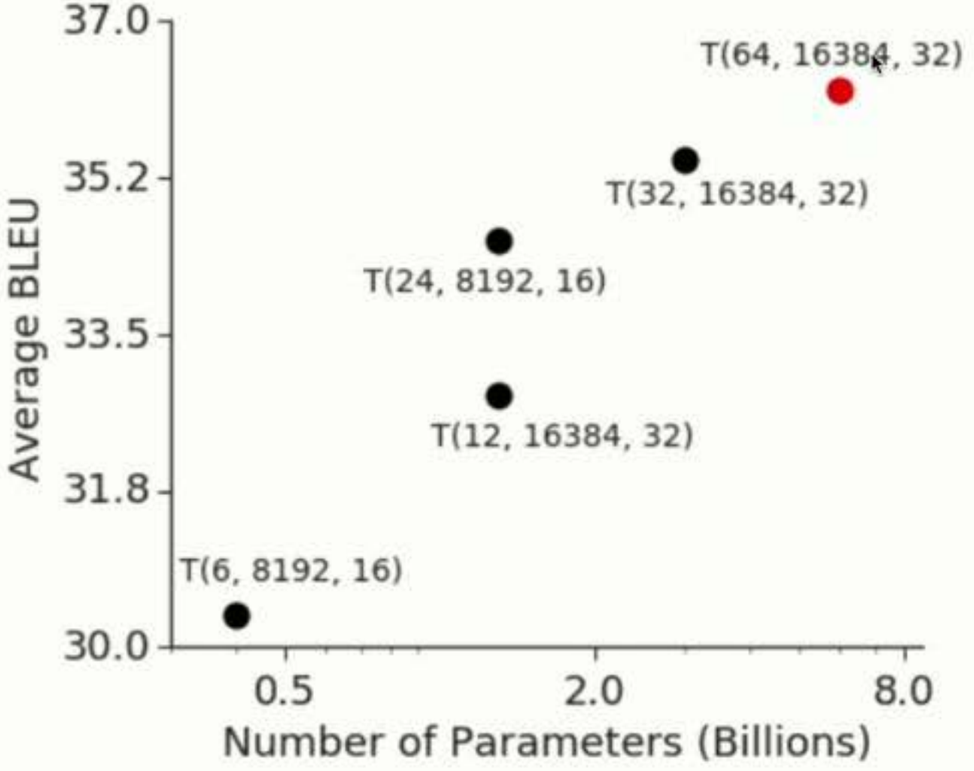
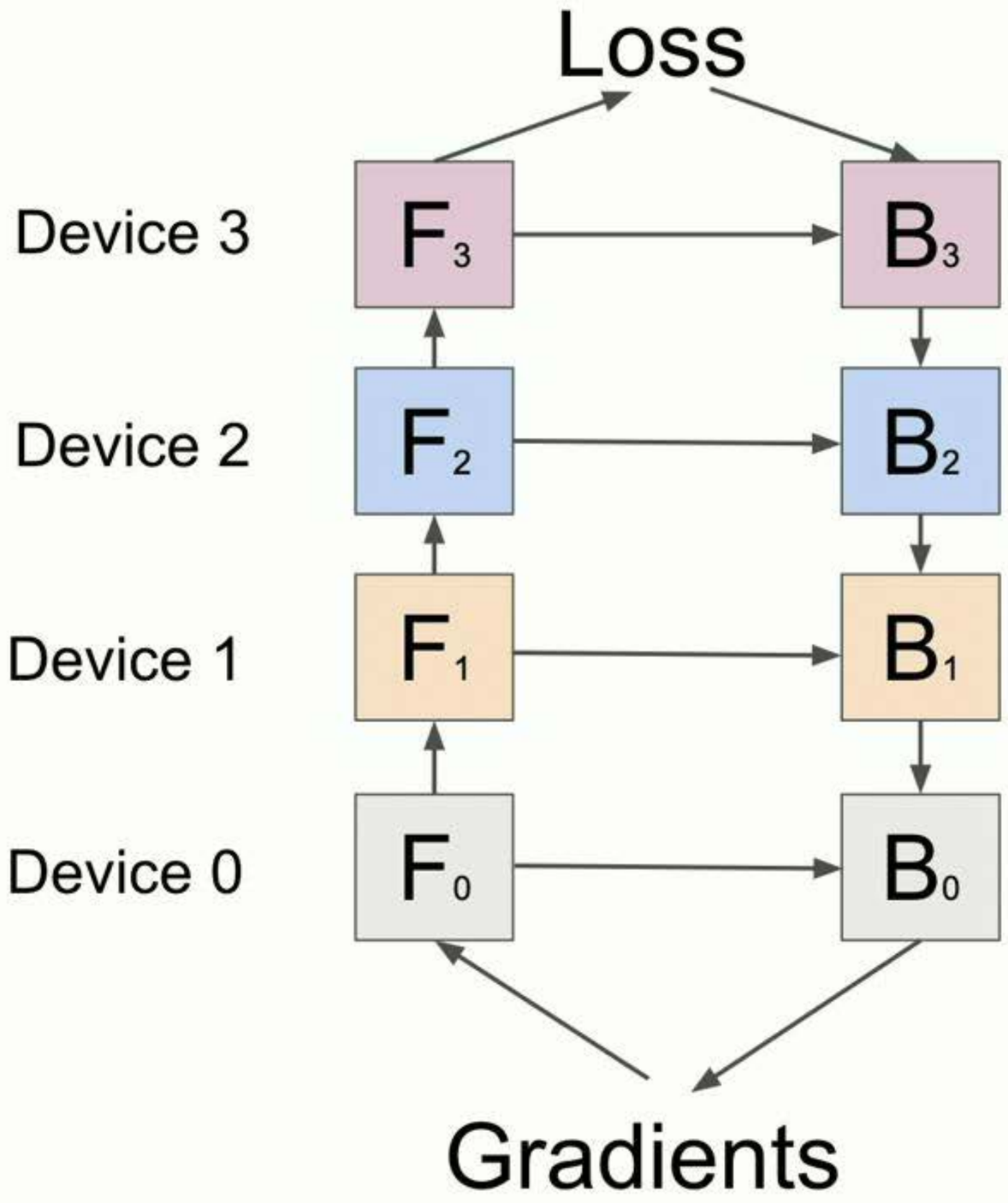


Image-Net



Machine Translation


Loss
Device 3
Device 2
Device 1
Device 0
F3
F2
F1
F0
B3
B2
B1
B0
Gradients

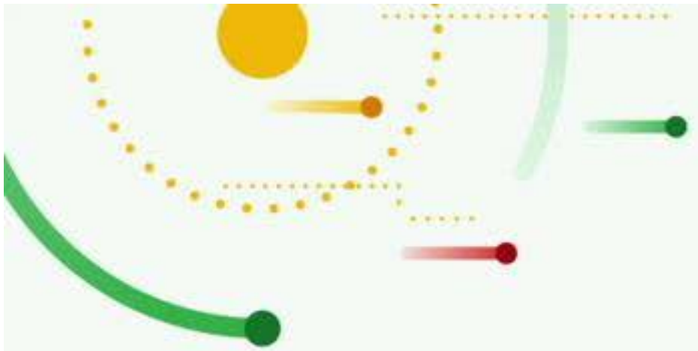
Performance

Sublinear scaling with or without high speed interconnect.

TPU	AmoebaNet			Transformer		
$K =$	2	4	8	2	4	8
$M = 1$	1	1.13	1.38	1	1.07	1.3
$M = 4$	1.07	1.26	1.72	1.7	3.2	4.8
$M = 32$	1.21	1.84	3.48	1.8	3.4	6.3

GPU	AmoebaNet			Transformer		
$K =$	2	4	8	2	4	8
$M = 32$	1	1.7	2.7	1	1.8	3.3

K: # of model partitions.
M:# of batch splitting

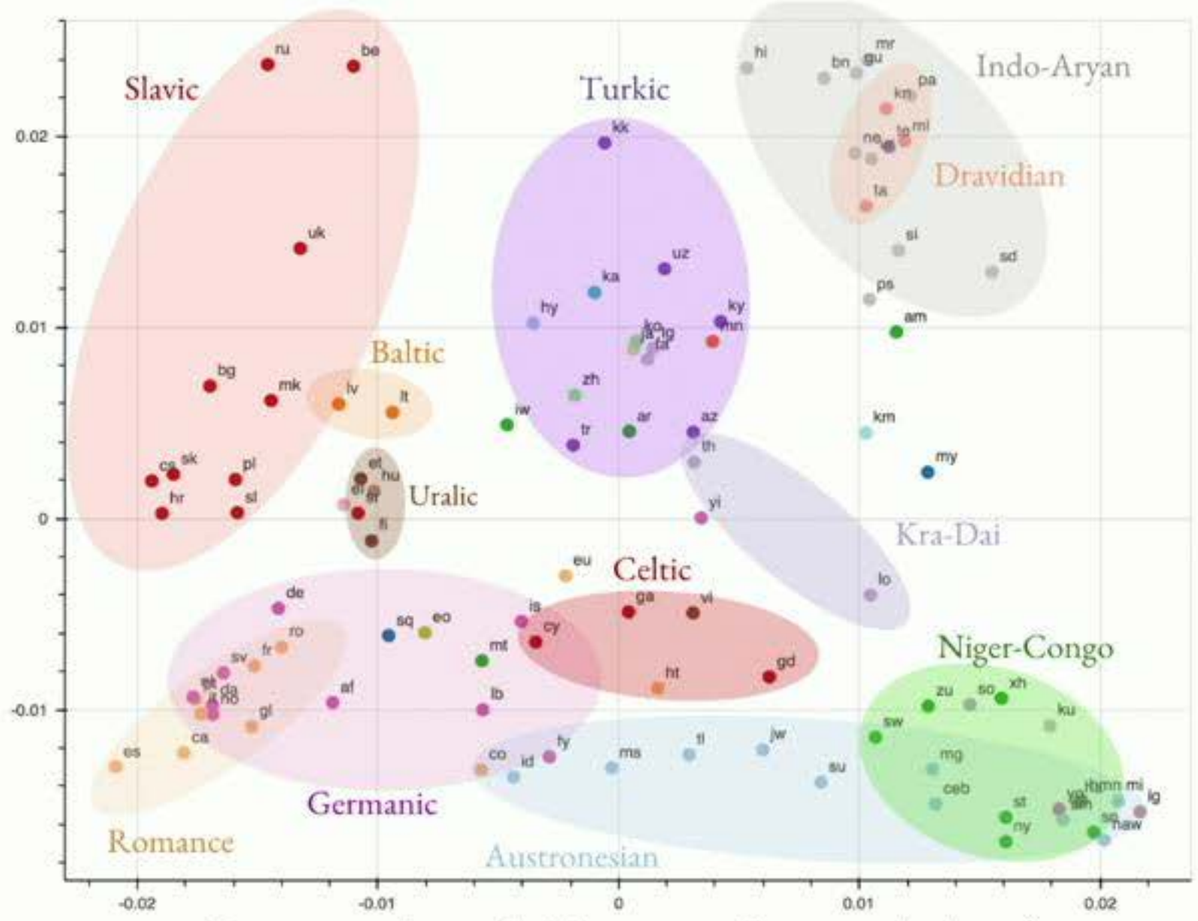


Google AI

Drawing the Map of Languages

Investigating Multilingual NMT Representations at Scale, Kudugunta et al. - EMNLP'19

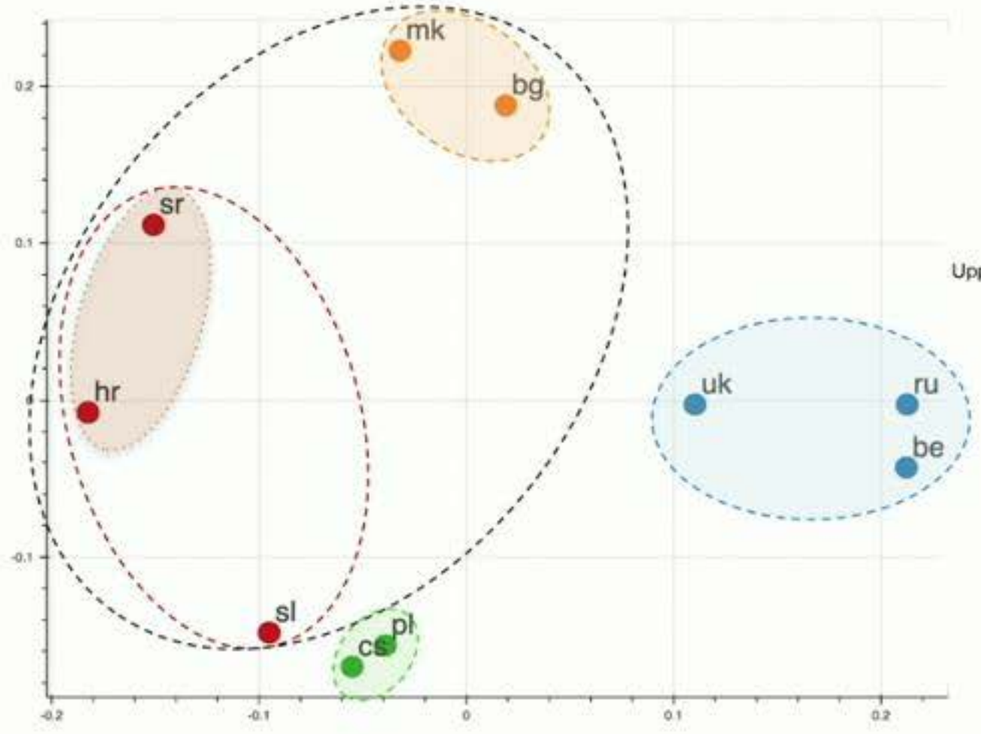
If we plot M4 language representations, they cluster based on linguistic similarity

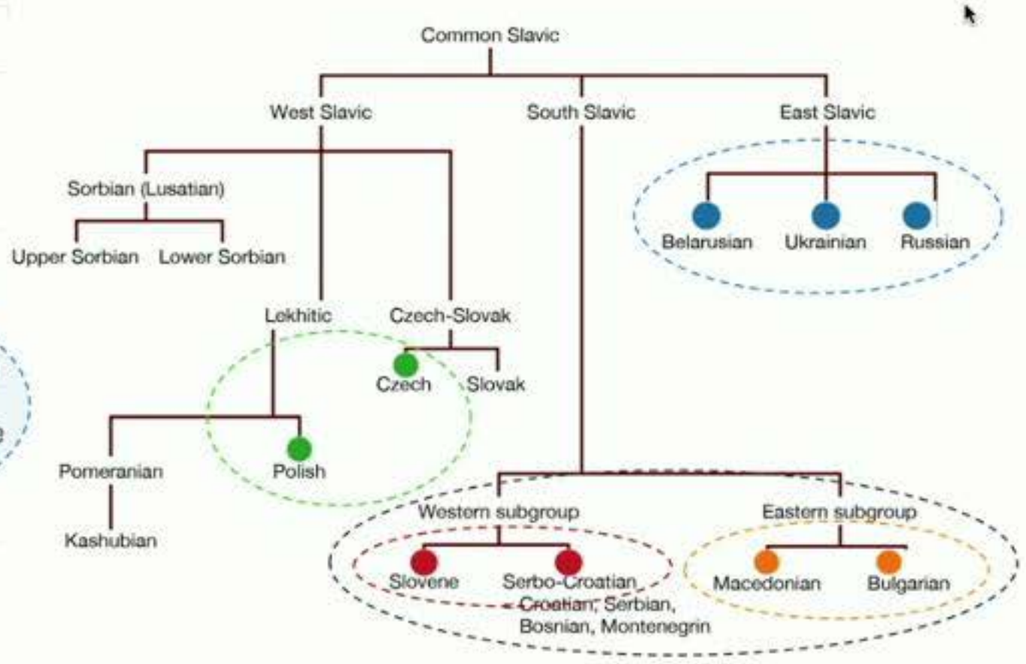


Representations of all languages (top encoder layer)

More details: Investigating Multilingual NMT Representations at Scale, Kudugunta et al. - EMNLP'19

When we look more closely into representations of language subfamilies, we also see a logical clustering





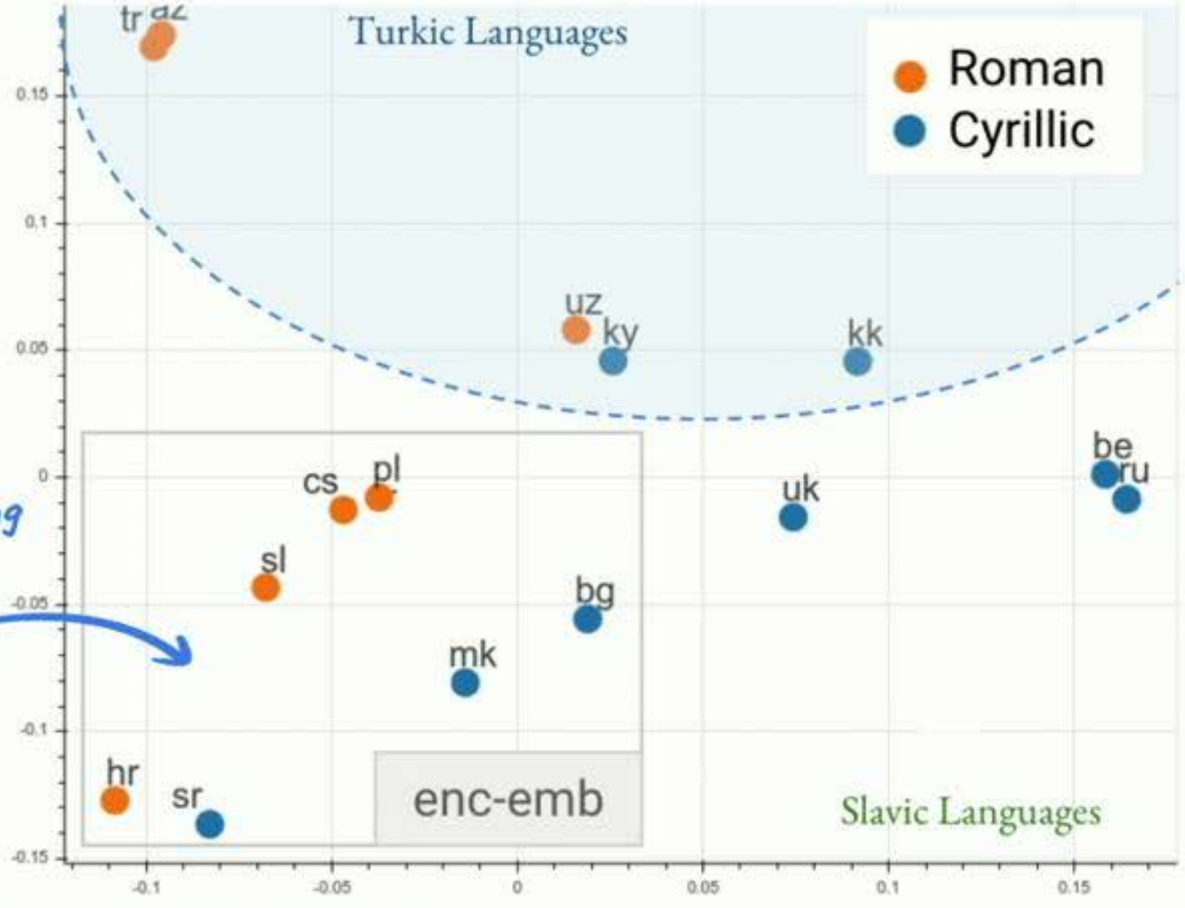
Slavic Languages (token embeddings).

More details: Investigating Multilingual NMT Representations at Scale, Kudugunta et al. - EMNLP'19

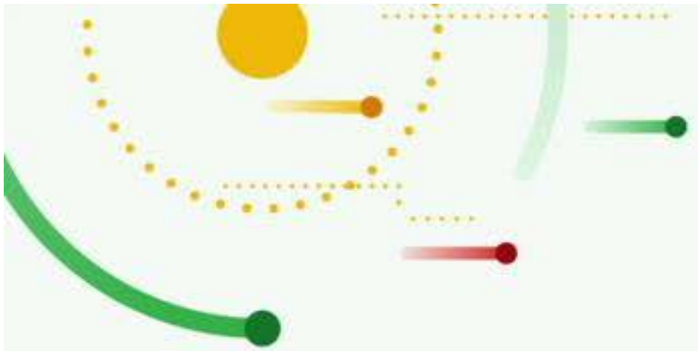
Linguistic similarity determines the representation affinity, especially in higher layers

Representations of Slavic and Turkic languages with Roman and Cyrillic scripts

Plot of encoder embedding representations (lowest layer)



More details: Investigating Multilingual NMT Representations at Scale, Kudugunta et al. - EMNLP'19

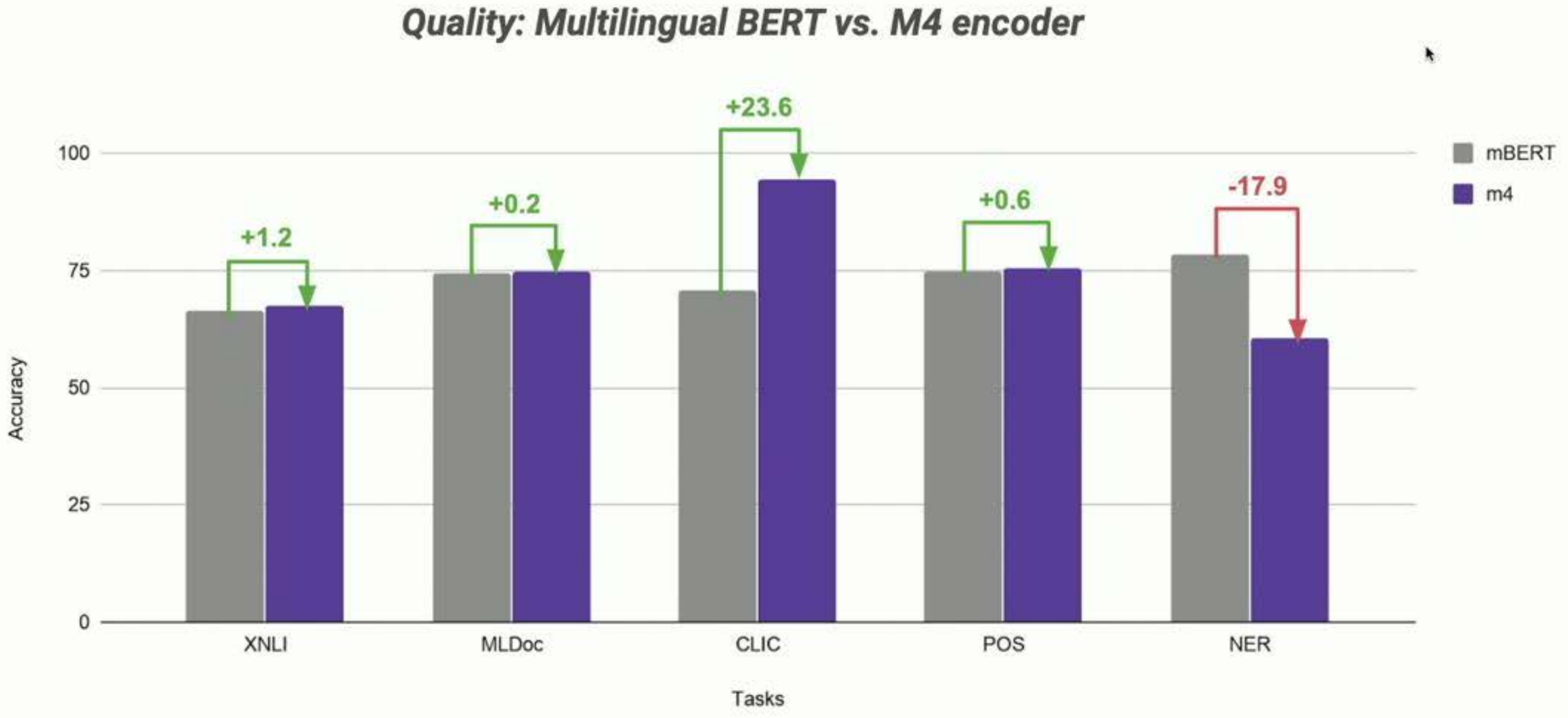


Google AI

Cross-lingual Downstream Transfer

Evaluating the Cross-Lingual Effectiveness of Massively Multilingual Neural Machine Translation,
Siddhant et al. AAAI 2020

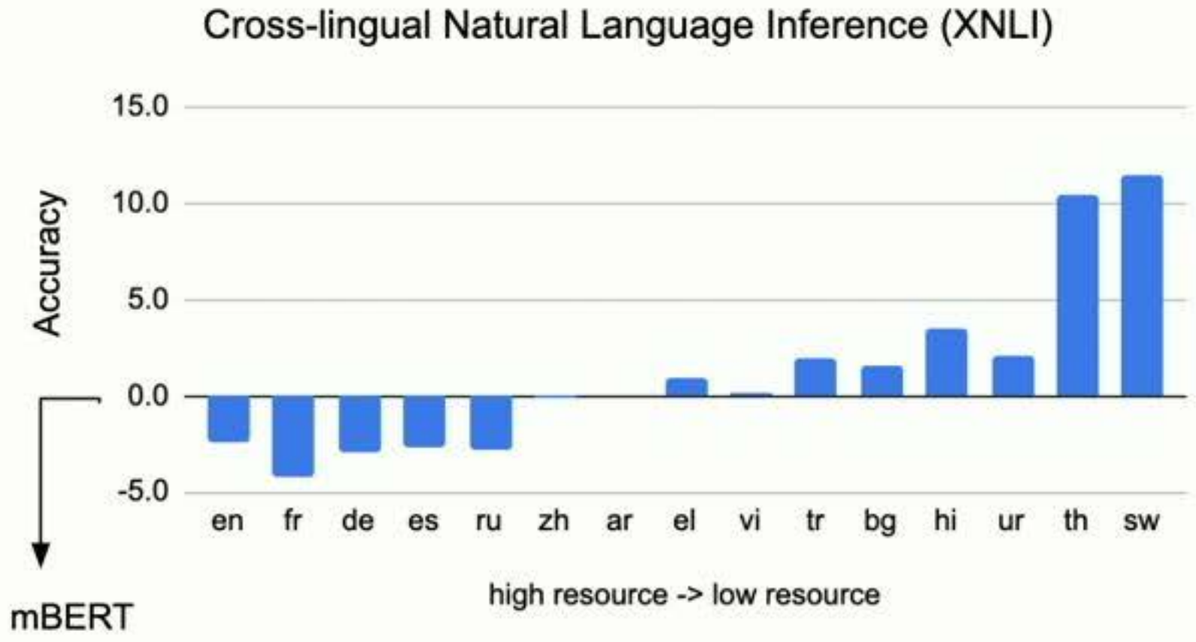
We evaluated M4 representations on downstream tasks; M4 encoder worked better than multilingual BERT in 4 out of 5 tasks

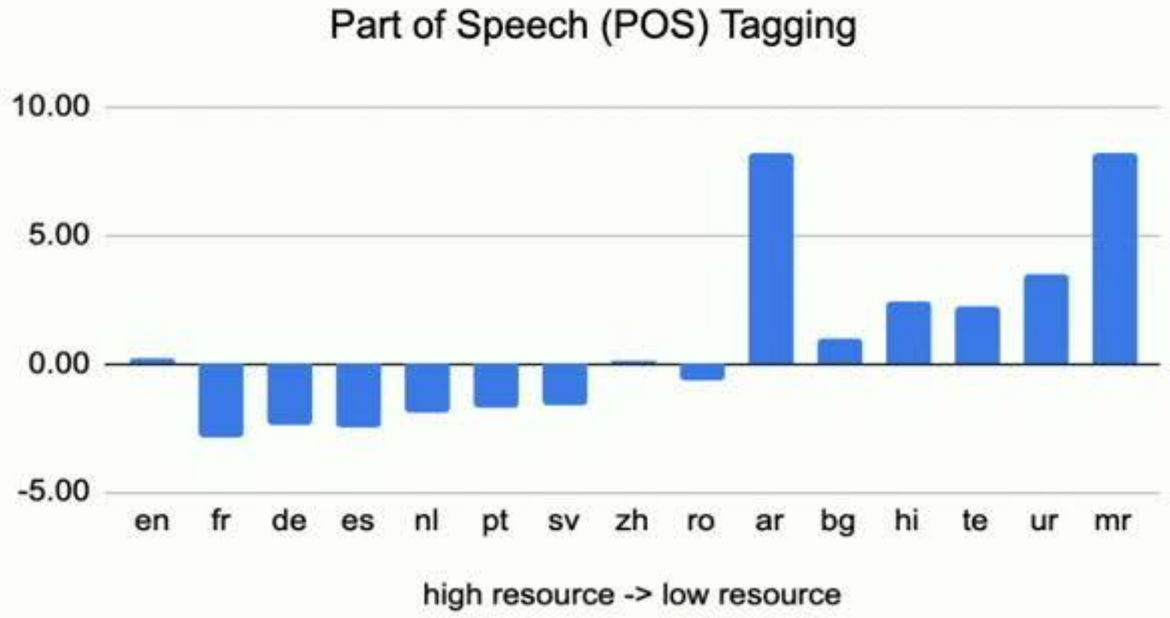


More details: Evaluating the Cross-Lingual Effectiveness of Massively Multilingual Neural Machine Translation, Siddhant et al. - AAAI 2020

M4 representations transfer to low-resource languages better than to high-resource languages

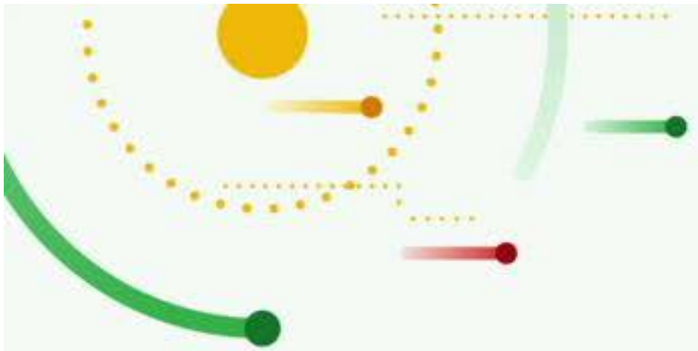
Quality: Multilingual BERT vs. M4 encoder





Hypotheses why transfer is better for low-resource languages:

- Translation to English from all languages implicitly forces their representation to be in the same space
- High resource languages are harder to move from point of convergence on fine-tuning

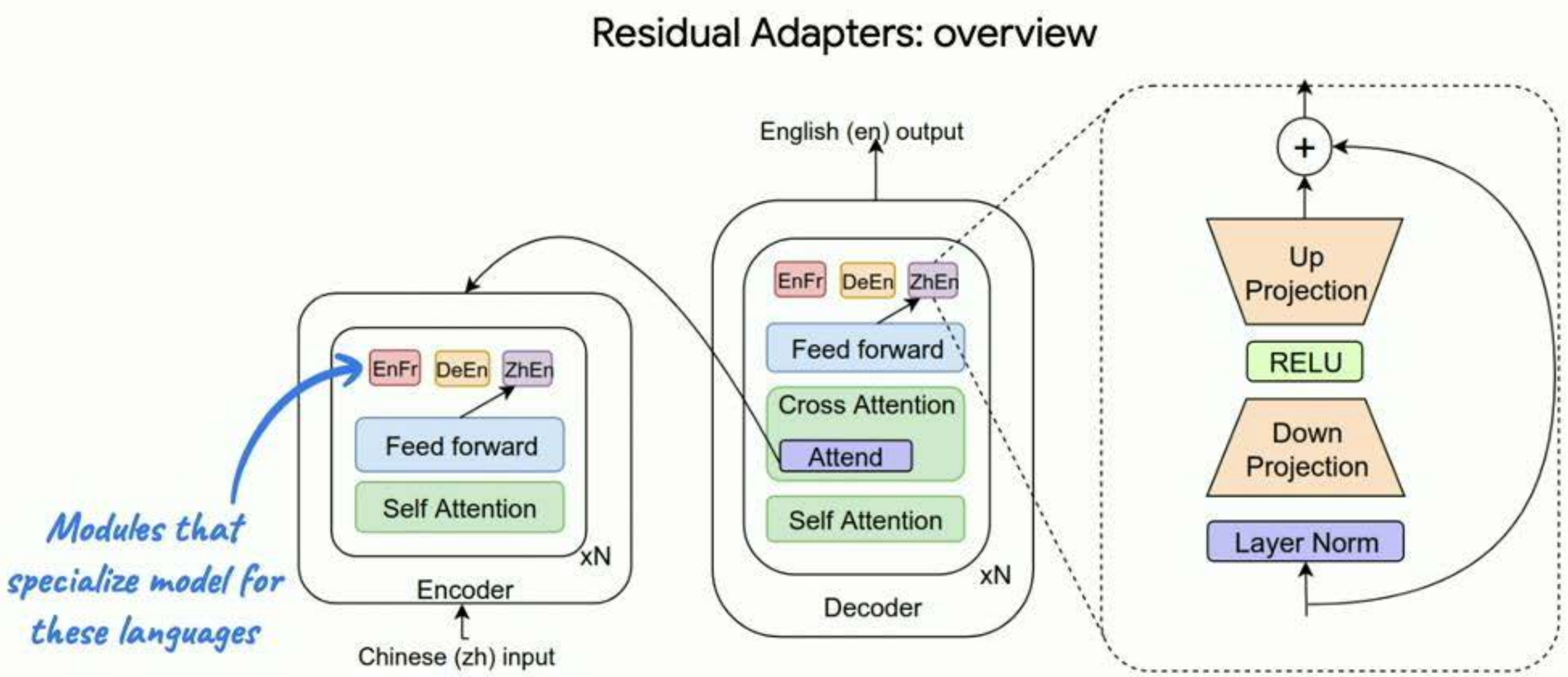


Google AI

Making M4 Practical: Simple Adaptation

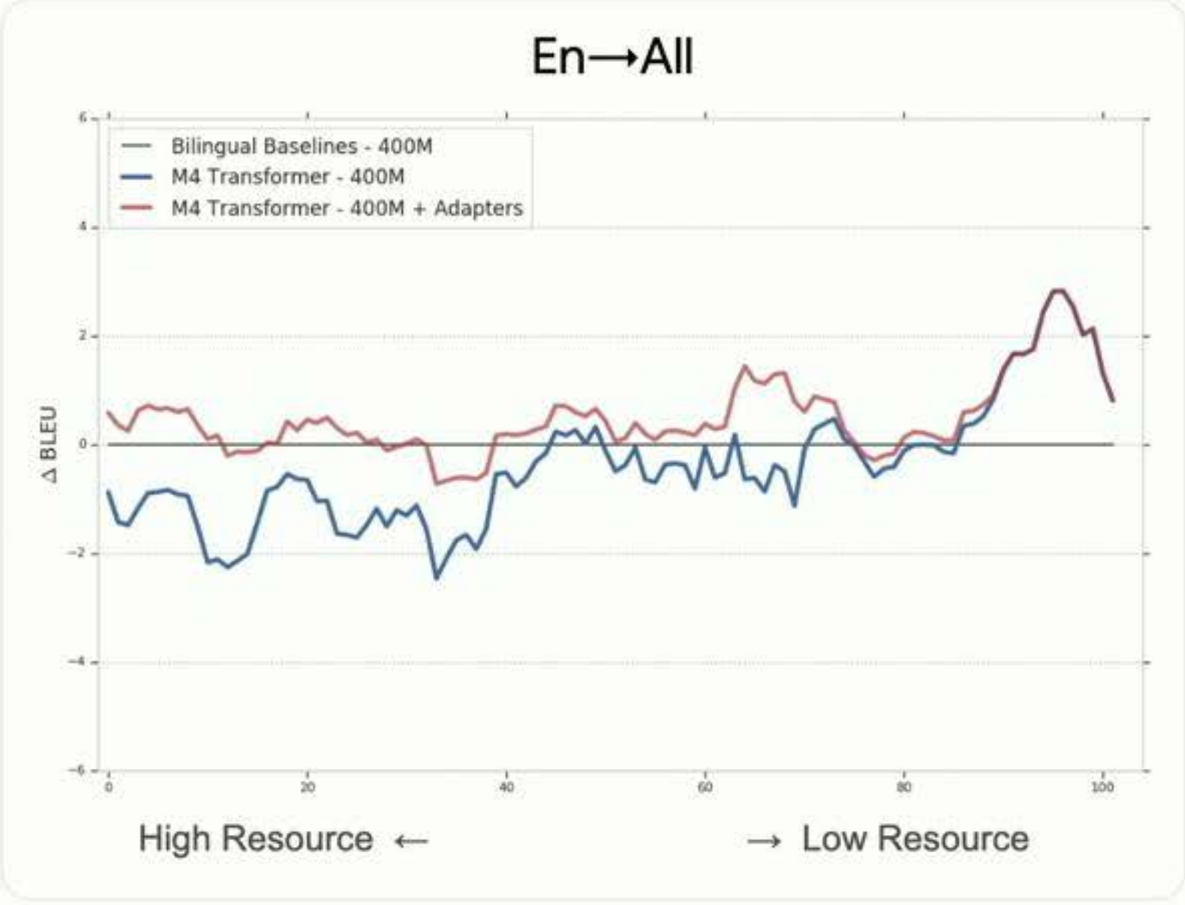
Simple, Scalable Adaptation for Neural Machine Translation, Bapna et al. - EMNLP'19

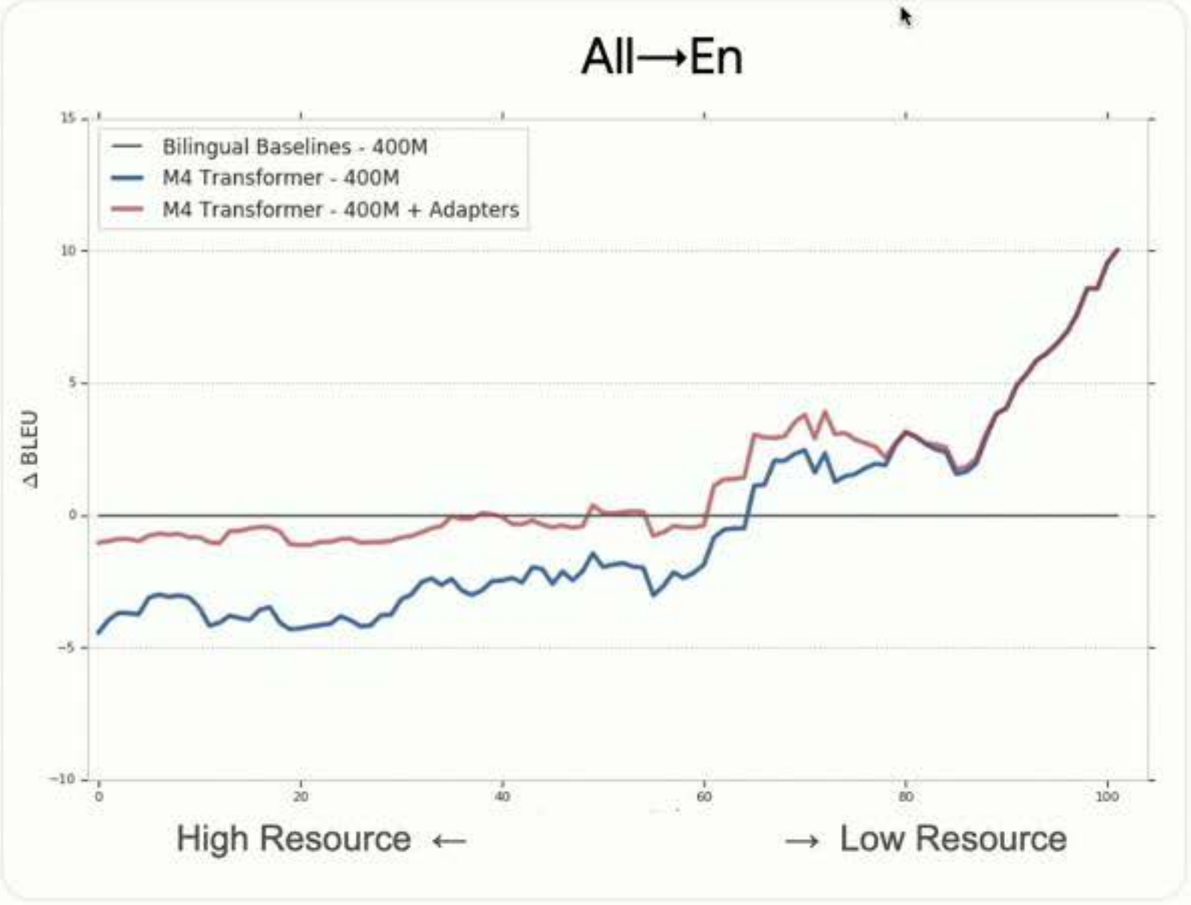
Residual Adapters allow for improving quality and for specializing models on languages or domains



Relevant paper: Simple, Scalable Adaptation for Neural Machine Translation (Bapna et al. - EMNLP'19)

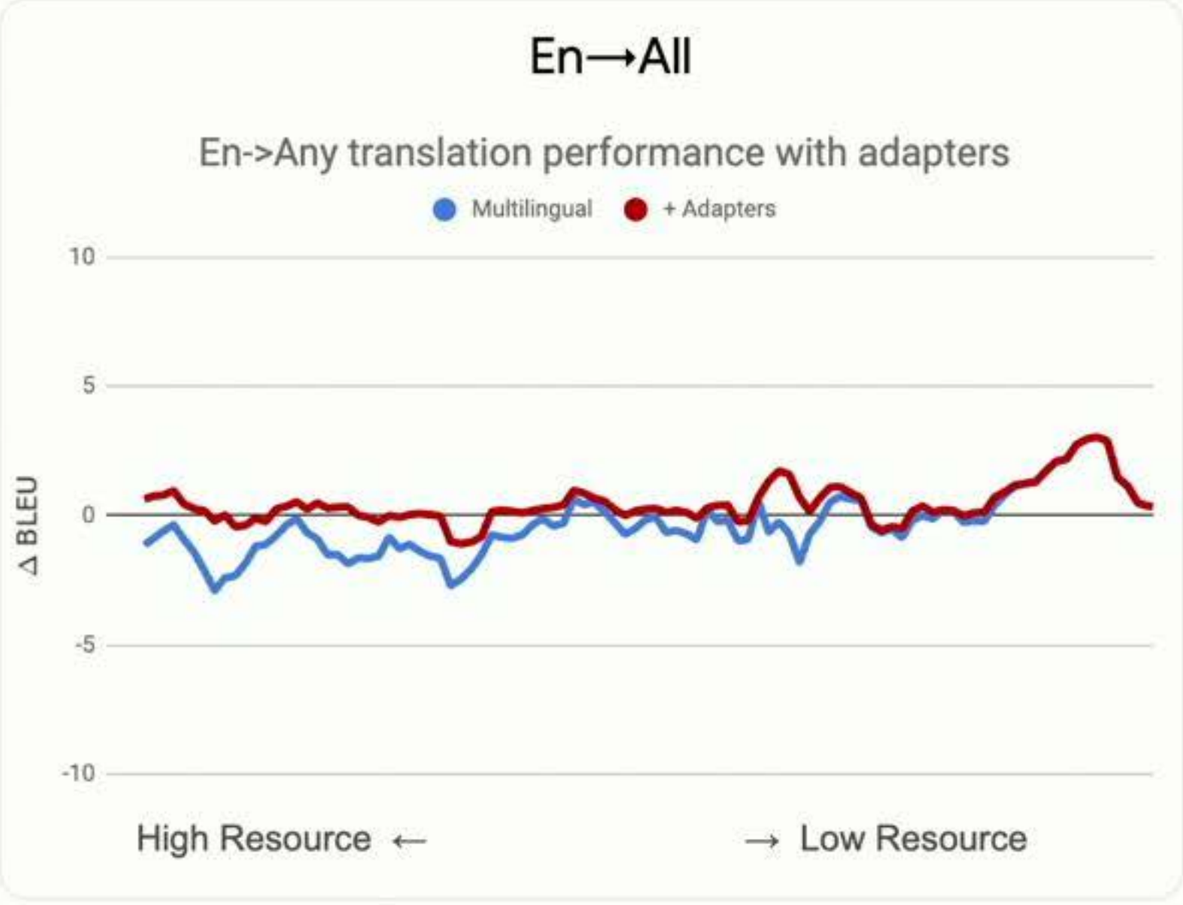
With Residual Adapters, we regain the quality drop in high-to-mid resource languages and enable domain-adaptation





Relevant paper: Simple, Scalable Adaptation for Neural Machine Translation (Bapna et al. - EMNLP'19)

Restore quality on high-to-mid resource languages with 13% extra parameters per language



Relevant paper: Simple, Scalable Adaptation for Neural Machine Translation (Bapna et al. - EMNLP'19)

For high resource languages we see further gains by increasing adapter capacity, although the returns are diminishing

Relevant paper: Simple, Scalable Adaptation for Neural Machine Translation (Bapna et al. - EMNLP'19)

We also summarized Phase 1 findings in a Google AI blog post (link)

Google AI Blog

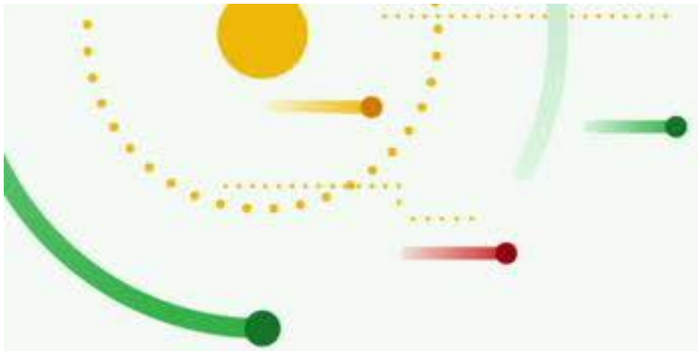
The latest news from Google AI

Exploring Massively Multilingual, Massive Neural Machine Translation

Friday, October 11, 2019

Posted by Ankur Bapna, Software Engineer and Orhan Firat, Research Scientist, Google Research

"... perhaps the way [of translation] is to descend, from each language, down to the common base of human communication — the real but as yet undiscovered universal language — and then re-emerge by whatever particular route is convenient." — Warren Weaver, 1949



Google AI

Enhancing Zero-Shot Translation:
The Missing Ingredient

The missing ingredient in zero-shot Neural Machine Translation, Arivazhagan et al. 2019.

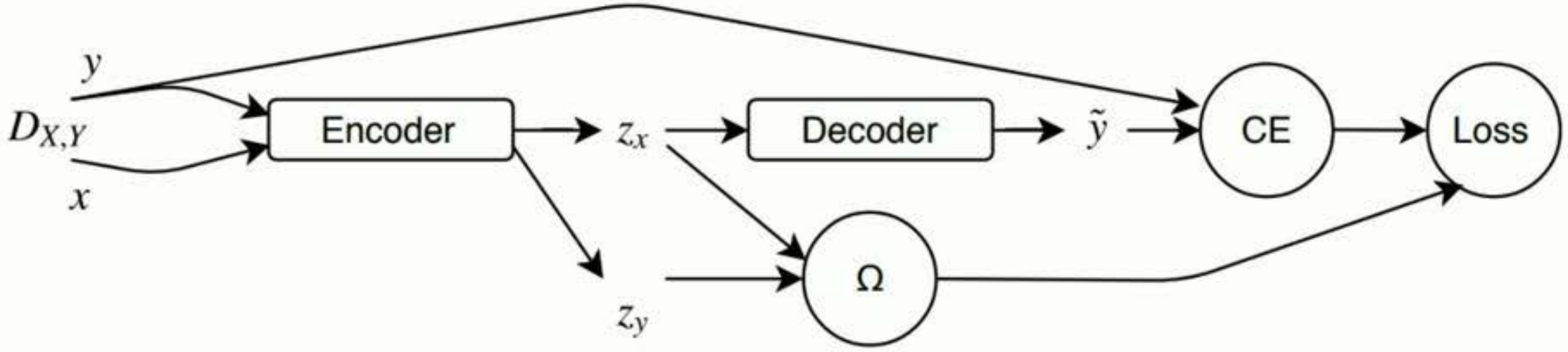
Enhancing Zero-Shot Translation: The Missing Ingredient

Bridging the gap between pivoting and zero-shot

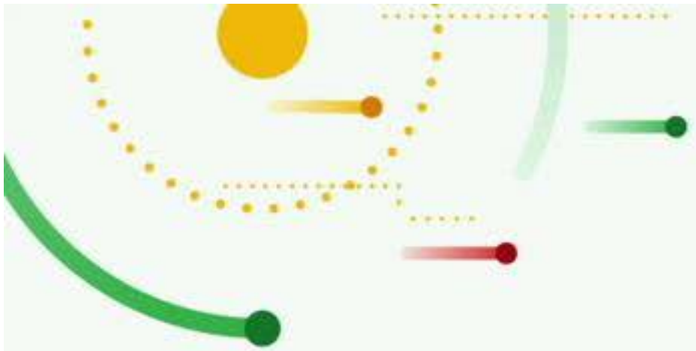
- auxiliary losses on the NMT encoder
- impose representational invariance via loss
- easy scalability to multiple languages

Group	vanilla		align(cosine)
	direct	pivot	direct
$en \leftrightarrow xx$ (8)	30.11	-	29.95
$xx \leftrightarrow yy$ (12)	16.73 (zs)	17.76	17.72 (zs)
All (20)	22.2	22.81	22.72

Table 5: Average BLEU scores for multilingual model on IWSLT-2017; Zero-Shot results are marked (zs).



The missing ingredient in zero-shot Neural Machine Translation, Arivazhagan et al. 2019.



Google AI

On training and hyper-parameters

Adaptive Schedules have the potential to overcome task-scheduling challenges

- **Meta-Learning** (for parallel transfer)
  - Learning to Schedule Tasks
    - Learning Task Weights [paper]
  - Explicit schedules: not scalable

$$w_i = 1 \Big/ \left(\min\left(1, \frac{s_i}{b_i}\right)^{\alpha} + \epsilon \right)$$

  - Implicit schedules: coupled with optimization

$$w_i = 1 + (\text{sign}\,(\overline{S} - S_i)) \min\left(\gamma, \left(\max_j S_j\right)^{\alpha} |S_i - \overline{S}|^{\beta}\right)$$

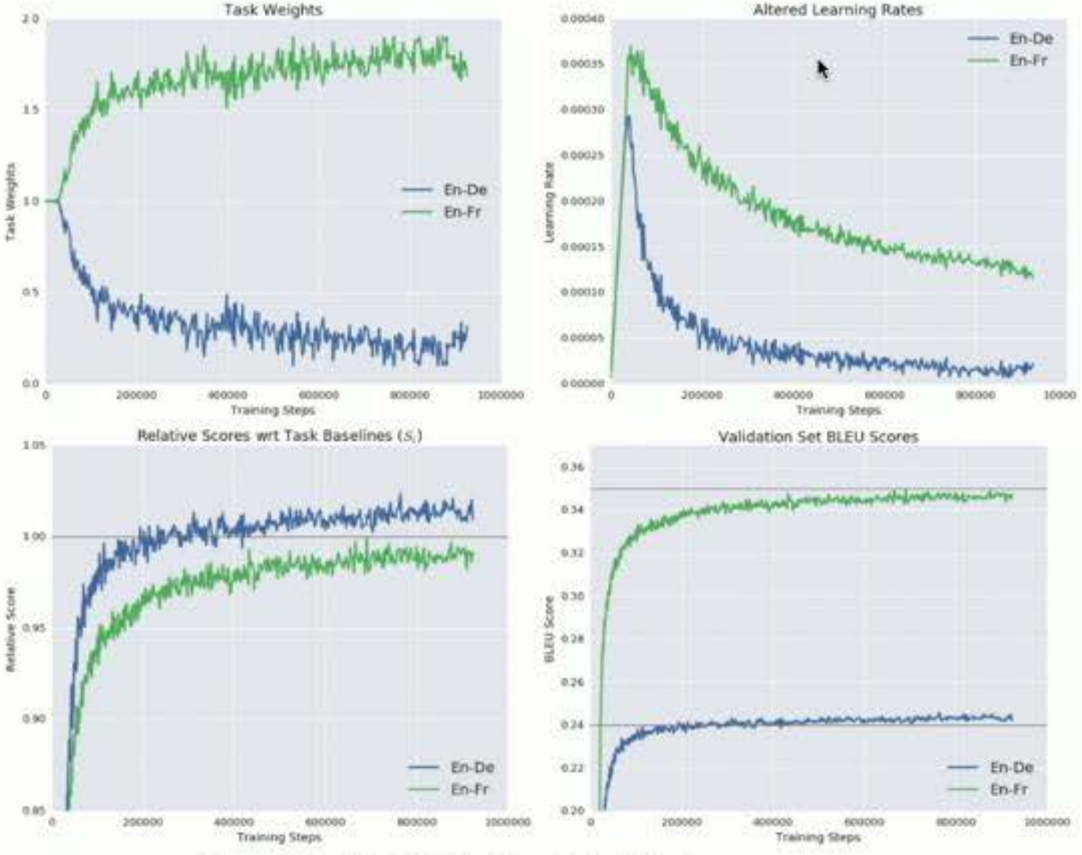


Figure 5: Implicit Validation-Based Scheduling Progress.

Adaptive Scheduling for Multi-task Learning, Jean et al. NeurIPS CLW 2018

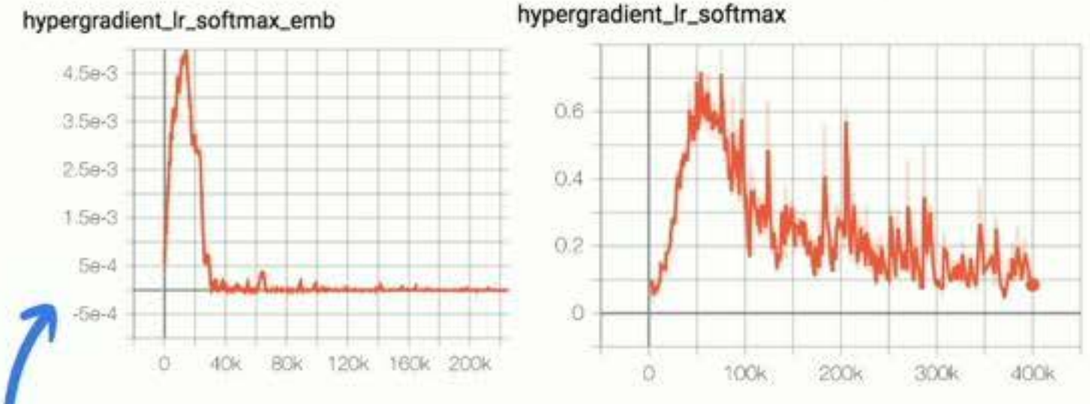
If we meta-learn some hyper-parameters, they don't require further tuning at scale

- Apply gradient descent on the learning rate (+underlying optimizer)

$$\frac{\partial f(\theta_{t-1})}{\partial \alpha} = \nabla f(\theta_{t-1}) \cdot \frac{\partial (\theta_{t-2} - \alpha \nabla f(\theta_{t-2}))}{\partial \alpha} = \nabla f(\theta_{t-1}) \cdot (-\nabla f(\theta_{t-2}))$$

$$\alpha_t = \alpha_{t-1} - \beta \frac{\partial f(\theta_{t-1})}{\partial \alpha} = \alpha_{t-1} + \beta \nabla f(\theta_{t-1}) \cdot \nabla f(\theta_{t-2})$$

- Comparison
  - Single pair (wmt'19 en-de): HG ~ Baseline
  - Multi-task (wmt en-{de,fr}): HG > Baseline
  - BERT: HG ~ Baseline



Learnt learning rate schedules (per-layer)

Hypergradients, Baydin et al. ICLR 2017

Adaptive Schedules have the potential to overcome task-scheduling challenges

- **Meta-Learning** (for parallel transfer)
  - Learning to Schedule Tasks
    - Learning Task Weights [paper]
  - Explicit schedules: not scalable

$$w_i = 1 \Big/ \left(\min\left(1, \frac{s_i}{b_i}\right)^{\alpha} + \epsilon \right)$$

  - Implicit schedules: coupled with optimization

$$w_i = 1 + (\text{sign}\,(\overline{S} - S_i)) \min\left(\gamma, \left(\max_j S_j\right)^{\alpha} |S_i - \overline{S}|^{\beta}\right)$$

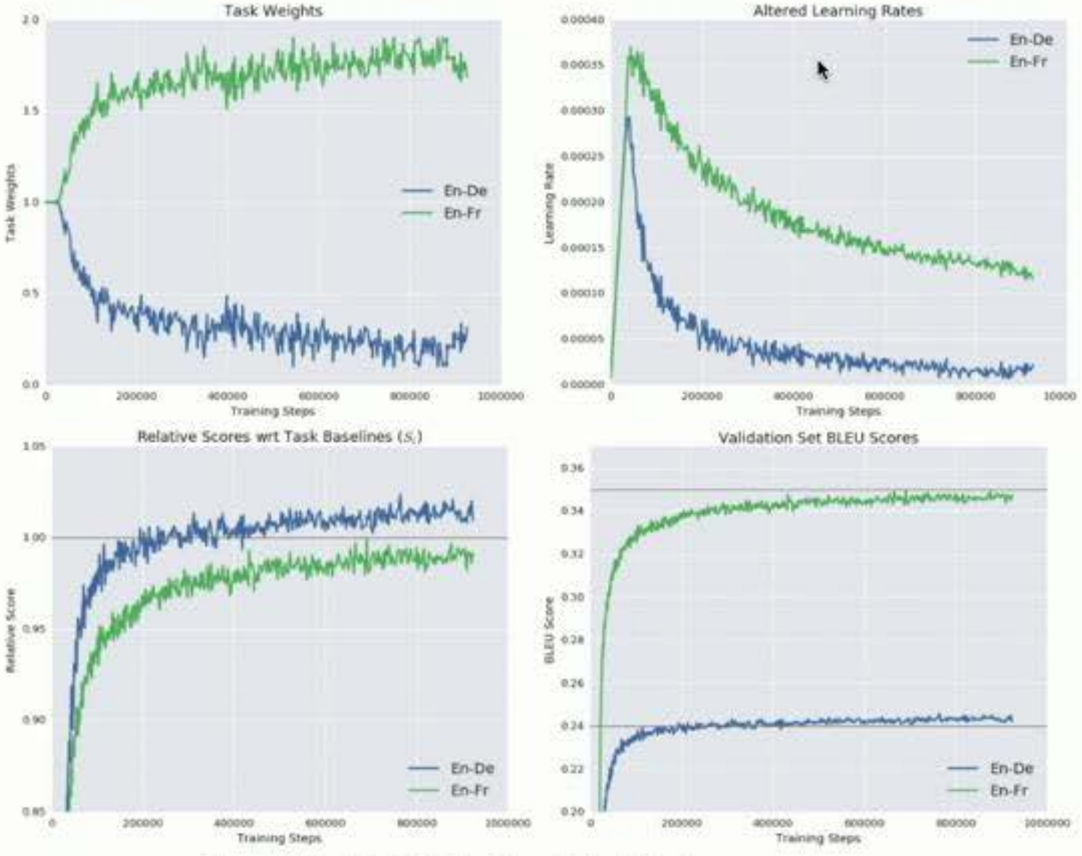


Figure 5: Implicit Validation-Based Scheduling Progress.

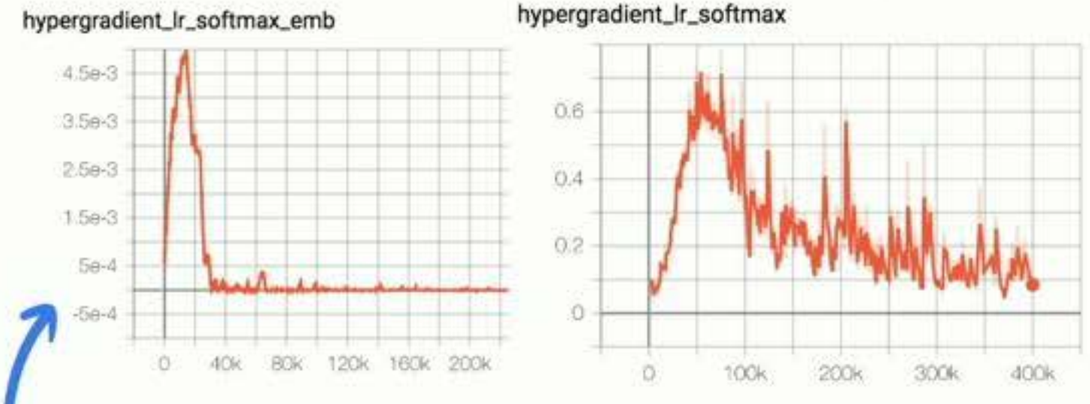
If we meta-learn some hyper-parameters, they don't require further tuning at scale

- Apply gradient descent on the learning rate (+underlying optimizer)

$$\frac{\partial f(\theta_{t-1})}{\partial \alpha} = \nabla f(\theta_{t-1}) \cdot \frac{\partial (\theta_{t-2} - \alpha \nabla f(\theta_{t-2}))}{\partial \alpha} = \nabla f(\theta_{t-1}) \cdot (-\nabla f(\theta_{t-2}))$$

$$\alpha_t = \alpha_{t-1} - \beta \frac{\partial f(\theta_{t-1})}{\partial \alpha} = \alpha_{t-1} + \beta \nabla f(\theta_{t-1}) \cdot \nabla f(\theta_{t-2})$$

- Comparison
  - Single pair (wmt'19 en-de): HG ~ Baseline
  - Multi-task (wmt en-{de,fr}): HG > Baseline
  - BERT: HG ~ Baseline



Learnt learning rate schedules (per-layer)

Hypergradients, Baydin et al. ICLR 2017

Challenges & Open Problems

Cross-lingual Downstream Transfer Learning

- Better learning objectives
- New tasks and languages

Unsupervised Machine Translation

- M4 as a knowledge base for unsupervised MT
- Adapting to unseen languages (modalities)

Transfer Learning

- Parallel transfer: smarter schedulers, mitigating interference
- Serial transfer: continual/meta learning, maximize transfer w/o forgetting

Adaptive Schedules have the potential to overcome task-scheduling challenges

- **Meta-Learning** (for parallel transfer)
  - Learning to Schedule Tasks
    - Learning Task Weights [paper]
  - Explicit schedules: not scalable

$$w_i = 1 \Big/ \left(\min\left(1, \frac{s_i}{b_i}\right)^{\alpha} + \epsilon \right)$$

  - Implicit schedules: coupled with optimization

$$w_i = 1 + (\text{sign}\,(\overline{S} - S_i)) \min\left(\gamma, \left(\max_j S_j\right)^{\alpha} |S_i - \overline{S}|^{\beta}\right)$$

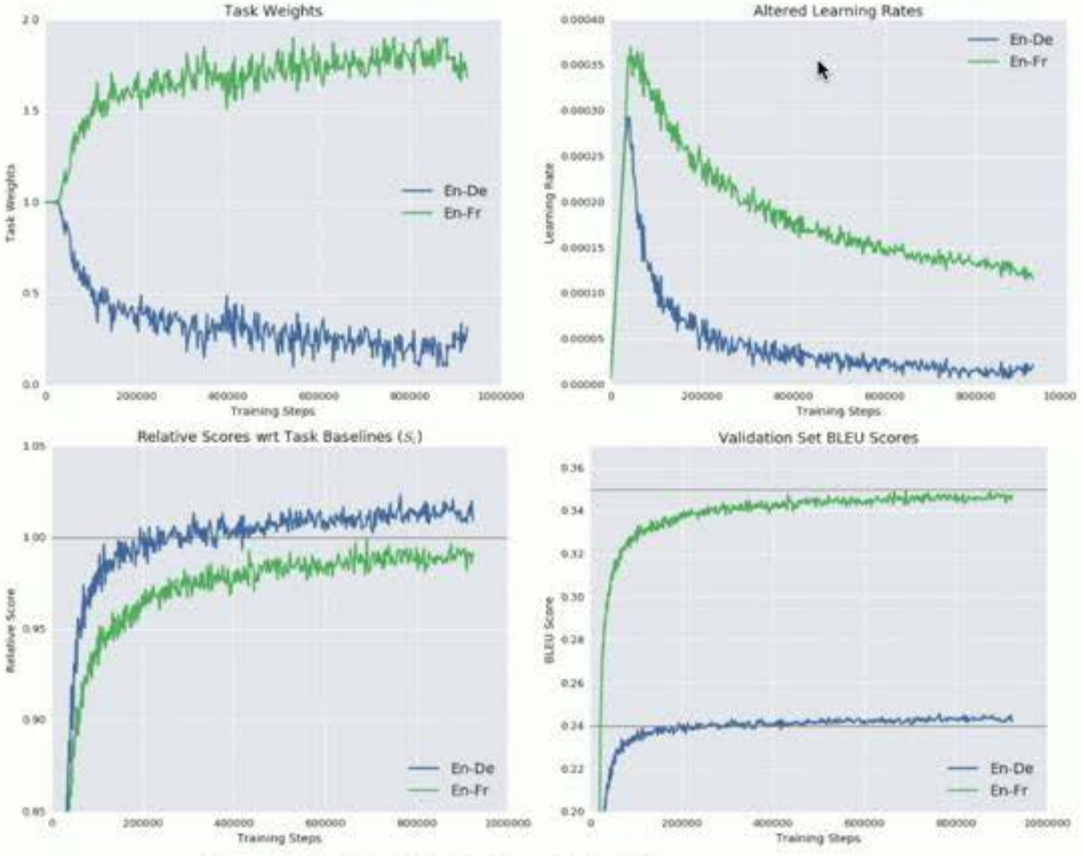


Figure 5: Implicit Validation-Based Scheduling Progress.

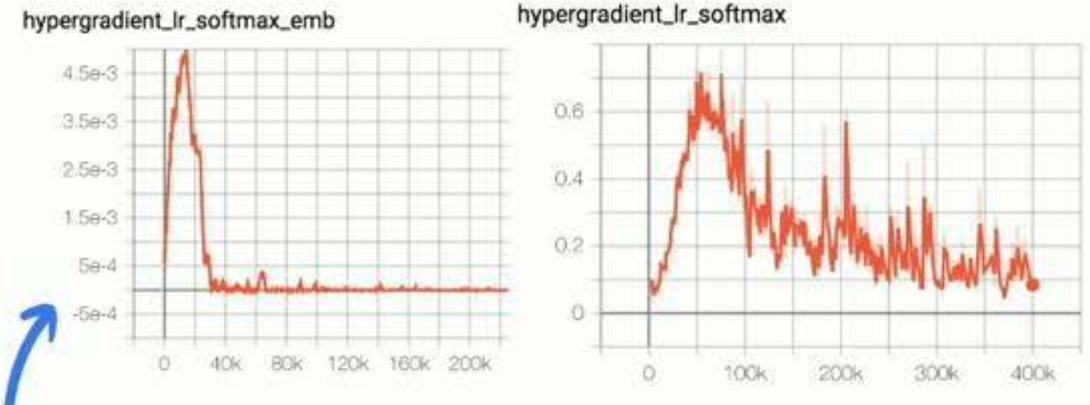
If we meta-learn some hyper-parameters, they don't require further tuning at scale

- Apply gradient descent on the learning rate (+underlying optimizer)

$$\frac{\partial f(\theta_{t-1})}{\partial \alpha} = \nabla f(\theta_{t-1}) \cdot \frac{\partial (\theta_{t-2} - \alpha \nabla f(\theta_{t-2}))}{\partial \alpha} = \nabla f(\theta_{t-1}) \cdot (-\nabla f(\theta_{t-2}))$$

$$\alpha_t = \alpha_{t-1} - \beta \frac{\partial f(\theta_{t-1})}{\partial \alpha} = \alpha_{t-1} + \beta \nabla f(\theta_{t-1}) \cdot \nabla f(\theta_{t-2})$$

- Comparison
  - Single pair (wmt'19 en-de): HG ~ Baseline
  - Multi-task (wmt en-{de,fr}): HG > Baseline
  - BERT: HG ~ Baseline

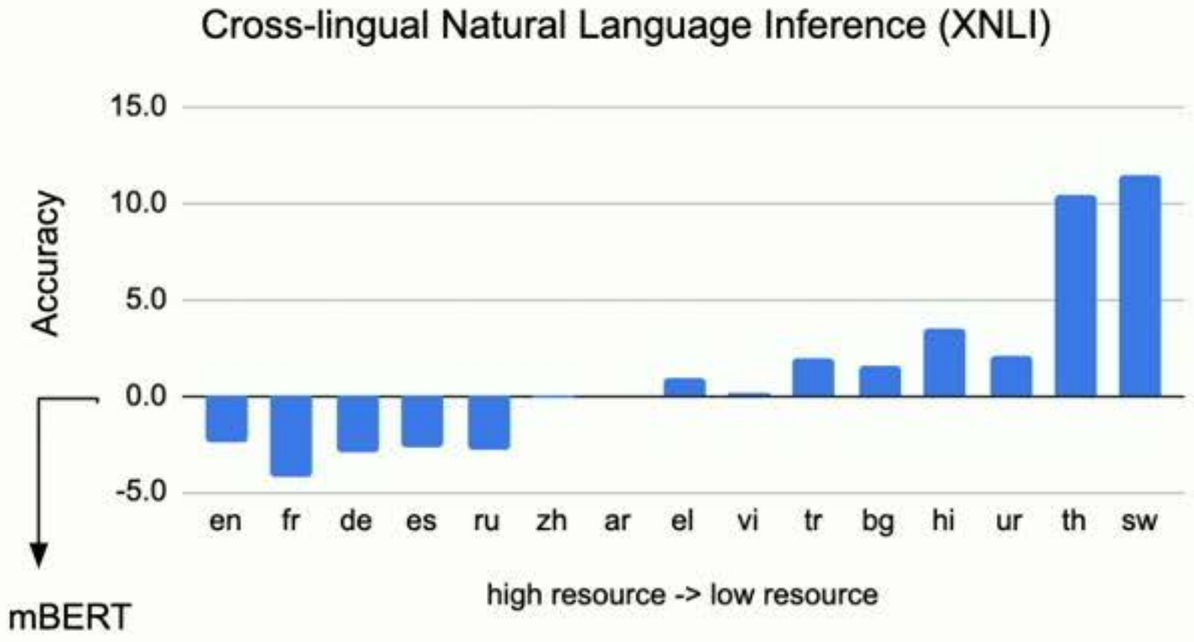


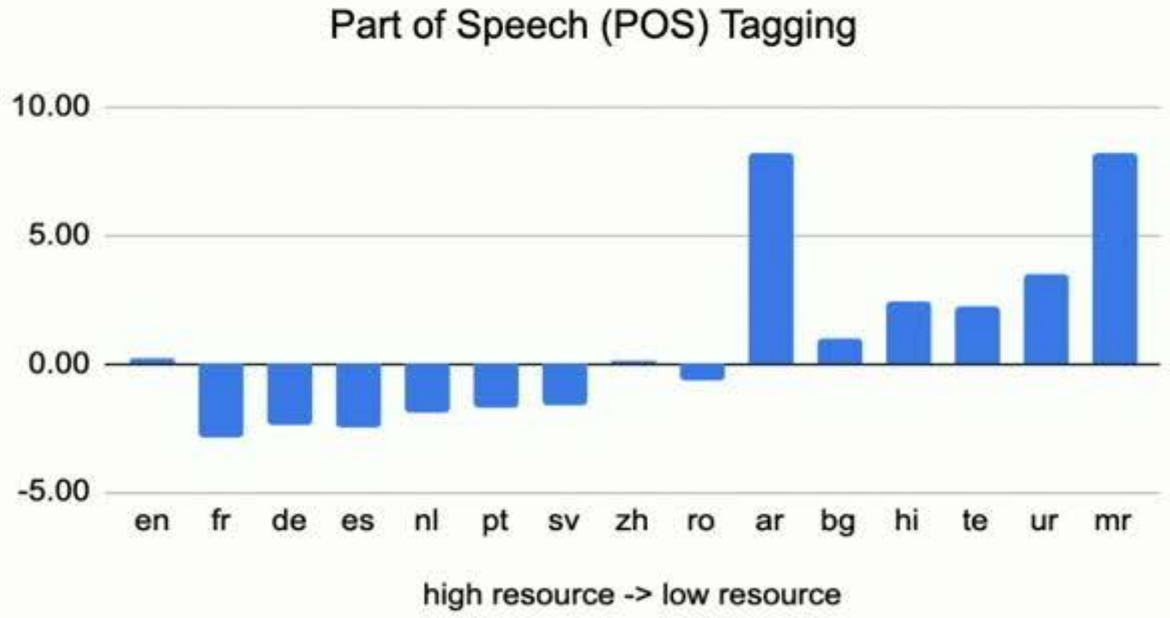
Learnt learning rate schedules (per-layer)

Hypergradients, Baydin et al. ICLR 2017

M4 representations transfer to low-resource languages better than to high-resource languages

Quality: Multilingual BERT vs. M4 encoder





Hypotheses why transfer is better for low-resource languages:

- Translation to English from all languages implicitly forces their representation to be in the same space
- High resource languages are harder to move from point of convergence on fine-tuning