Multilingual language models based on the transformer architecture

Why multilingual?

- **Languages other** than English or Russian do exist.
  - *And as empires fall apart, new languages get official status and wider usage*
  - *Users want their content in their own languages*

Why multilingual?

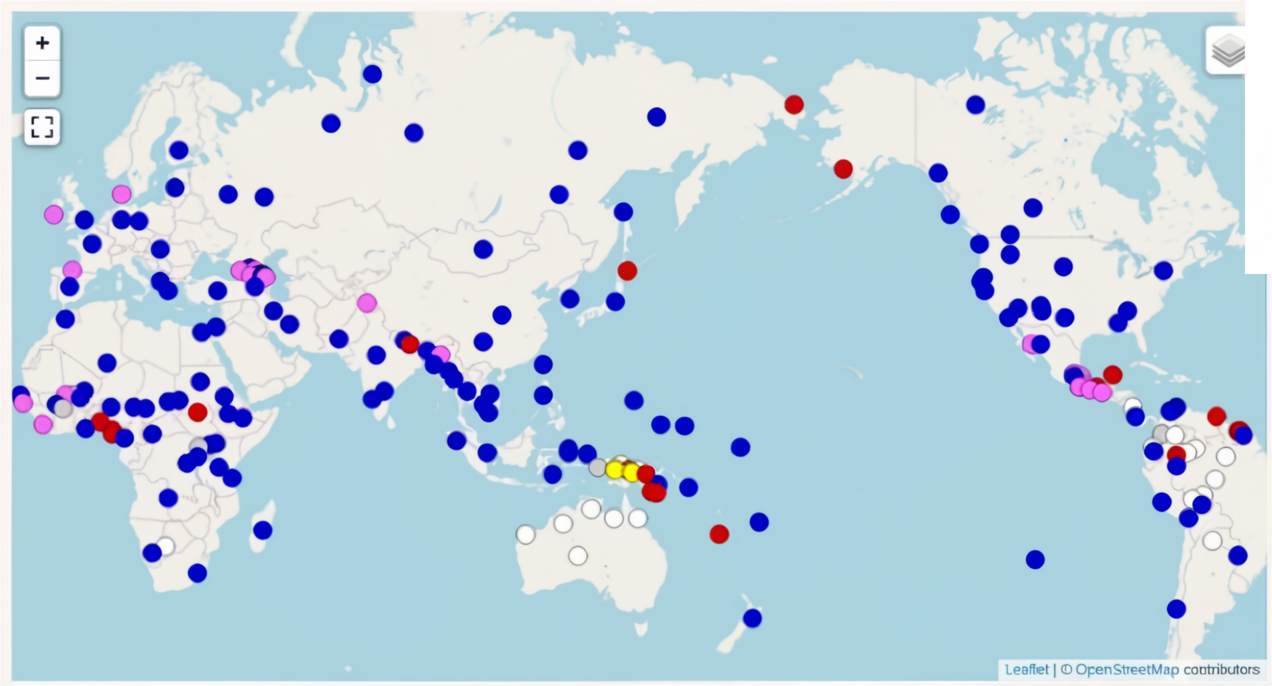
- **Languages other** than English or Russian do exist.
  - *And as empires fall apart, new languages get official status and wider usage*
  - *Users want their content in their own languages*
- It is **expensive** to support **separate NLP models** for each language

Why multilingual?

- **Languages other** than English or Russian do exist.
  - *And as empires fall apart, new languages get official status and wider usage*
  - *Users want their content in their own languages*
- It is **expensive** to support **separate NLP models** for each language
- Most languages are “low-resource”
  - *Monolingual models for them are often not good enough*
  - *But we can transfer NLP knowledge across languages*
    - For closely related languages (e.g. ru->by), it can be transferred directly
    - For more distant languages, translation might be required

The language space: WALS typology

The World Atlas of Language Structures stores unified language *features*



Values

🔵	Decimal	125
🟣	Hybrid vigesimal-decimal	22
🔴	Pure vigesimal	20
⚪	Other base	5
🟡	Extended body-part system	4
◯	Restricted	20

🔵 Синий	**Десятичная система** (Decimal)	125
🌸 Розовый	**Гибридная**: двадцатеричная + десятичная (Hybrid vigesimal-decimal)	22
🔴 Красный	**Чисто двадцатеричная** (Pure vigesimal)	20
⚪ Белый	**Ограниченная система** (Restricted)	20
🟡 Жёлтый	**Система, основанная на частях тела** (Extended body-part system)	4
⚫ Серый	**Другая основа** (Other base)	5

🗺️ **What the map shows:**

🔵 **Decimal systems** dominate in Europe, Asia, and large parts of Africa and the Americas.
🔴 **Pure vigesimal (base-20)** systems appear in specific regions like parts of Central America, Africa, and the Arctic.
🌸 **Hybrid vigesimal-decimal** systems are found in scattered areas, such as parts of Europe and Central America.
⚪ **Restricted numeral systems** (limited counting systems) occur mainly in regions with small or traditionally oral languages.
🟡 **Extended body-part systems** are rare and primarily found in areas like Papua New Guinea. locations.

http://phontron.com/class/multiling2022/assets/slides/multiling-03-typology.pdf ;
https://wals.info/

The language space: WALS typology

- 2600+ languages, 190+ attributes

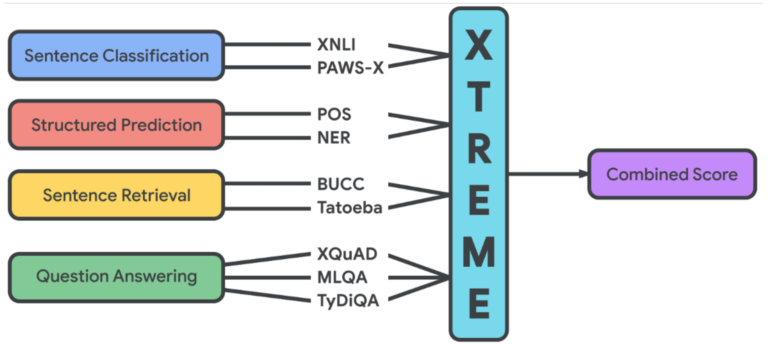
ID#	Feature Name	Category	Feature Values
1	Consonant Inventories	Phonology (19)	{1:Large, 2:Small, 3:Moderately Small, 4:Moderately Large, 5:Average}
23	Locus of Marking in the Clause	Morphology (10)	{1:Head, 2:None, 3:Dependent, 4:Double, 5:Other}
30	Number of Genders	Nominal Categories (28)	{1:Three, 2:None, 3:Two, 4:Four, 5:Five or More}
58	Obligatory Possessive Inflection	Nominal Syntax (7)	{1:Absent, 2:Exists}
66	The Perfect	Verbal Categories (16)	{1:None, 2:Other, 3:From 'finish' or 'already', 4:From Possessive}
81	Order of Subject, Object and Verb	Word Order (17)	{1:SVO, 2:SOV, 3:No Dominant Order, 4:VSO, 5:VOS, 6:OVS, 7:OSV}
121	Comparative Constructions	Simple Clauses (24)	{1:Conjoined, 2:Locational, 3:Particle, 4:Exceed}
125	Purpose Clauses	Complex Sentences (7)	{1:Balanced/deranked, 2:Deranked, 3:Balanced}
138	Tea	Lexicon (10)	{1:Other, 2:Derived from Sinitic 'cha', 3:Derived from Chinese 'te'}
140	Question Particles in Sign Languages	Sign Languages (2)	{1:None, 2:One, 3:More than one}
142	Para-Linguistic Usages of Clicks	Other (2)	{1:Logical meanings, 2:Affective meanings, 3:Other or none}

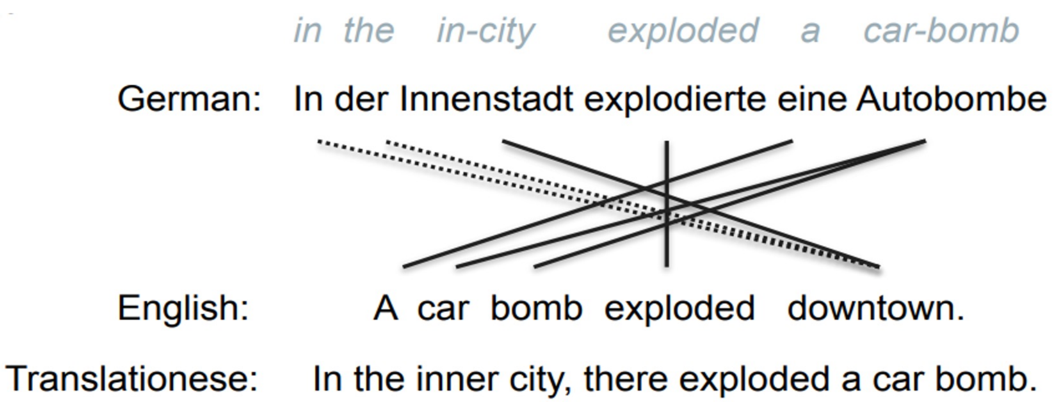
Example from Georgi, Xia and Lewis (2010)

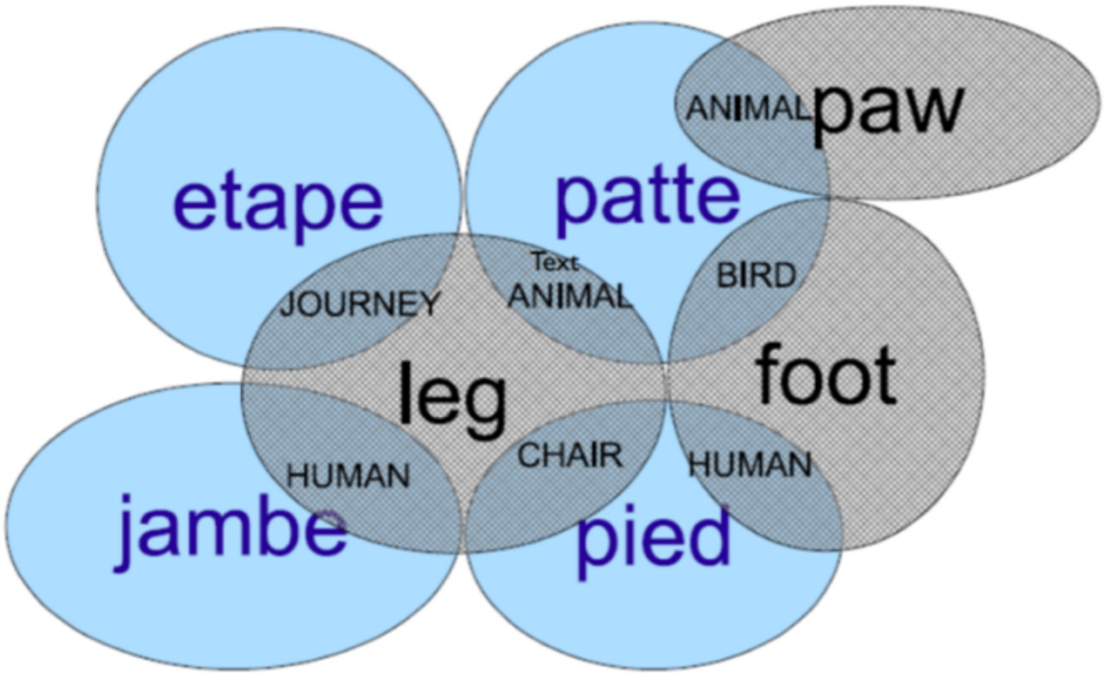
http://phontron.com/class/multiling2022/assets/slides/multiling-03-typology.pdf ; https://wals.info/

Examples of multilingual tasks

- Translation between multiple languages (e.g. FLORES(-200))
- MASSIVE NLU benchmark in 51 language from Amazon Alexa
  - Recognize intents and slots in dialogues with assistant in any language
- NeuCLIR benchmark in cross-language information retrieval
  - Search among Zh, Fa and Ru documents with En queries
- Multilingual News Article Similarity
- Multilingual Complex Named Entity Recognition
- Composite benchmarks: XTREME, XGLUE







📍 **NER: "Nadezhda"**
In Russian, the word **"Надежда"** can mean:
1. **A personal name** (female first name) → *Nadezhda*
2. **A location** (e.g., "Мыс Доброй Надежды" = *Cape of Good Hope*) → *Hope*
3. **A common noun** (*hope*) → *hope*

🤖 **What Can Go Wrong Without NER:**

All instances might be translated as *"hope"*:
•*I spoke with hope before going to the Cape of Good Hope to discuss hope...*

Or worse, all as a name:
•*I spoke with Nadezhda before the trip to Nadezhda to discuss Nadezhda...*

Эти типы стали есть на складе

- Материал находится на складе
- Люди едят на складе
- Сталь нужно есть на складе

Images source: https://web.stanford.edu/~jurafsky/slp3/

Examples of parallel corpora

- Important books
  - Bible, Tanzil (Quran)
- Governmental texts
  - Europarl, UN corpus, etc.
- Subtitles
  - OpenSubtitles, TED, etc.
- Computer manuals
  - PHP, Ubuntu, etc.
- Aligned web data
  - ParaCrawl, WikiMatrix, CCMatrix, etc.
- A major repository: OPUS

Multilingual models

How multilingual is multilingual BERT?

- The most typical multilingual pretrained models are BERT-like
  - E.g. multilingual BERT (2018), XLM(2019), XLM-R (2020), mDeBERTaV3 (2021)
- Most of them (except XLM) are fully unsupervised
- Still, they can perform cross-language transfer
- How does it even work???
  - Common vocabulary
  - Some mapping between vocabularies of similar languages
    - E.g. Hindi (Devanagari script) vs Urdu (Arabic script)
  - Generalization depends on the number of shared WALS features
- Perhaps, alignment occurs via shared words (e.g. URLs)

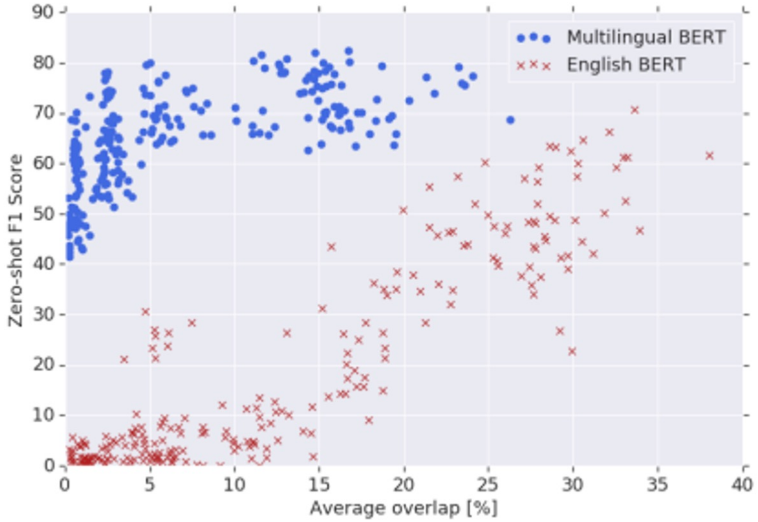


Figure 1: Zero-shot NER F1 score versus entity word piece overlap among 16 languages. While performance using EN-BERT depends directly on word piece overlap, M-BERT's performance is largely independent of overlap, indicating that it learns multilingual representations deeper than simple vocabulary memorization.

How multilingual is multilingual BERT? Pires et al, 2019, https://arxiv.org/abs/1906.01502

XLM-R(oBERTa)

XLM-R (XLM-RoBERTa)

- A single RoBERTa model trained on **2.5TB** of **filtered** texts from CommonCrawl on **100 languages**
- Shows zero-shot cross-lingual transfer ability!
- Takes best from XLM (multilingual) and RoBERTa (English) models

Unsupervised Cross-lingual Representation Learning at Scale

Alexis Conneau* **Kartikay Khandelwal***

Naman Goyal **Vishrav Chaudhary** **Guillaume Wenzek** **Francisco Guzmán**

Edouard Grave **Myle Ott** **Luke Zettlemoyer** **Veselin Stoyanov**

Facebook AI

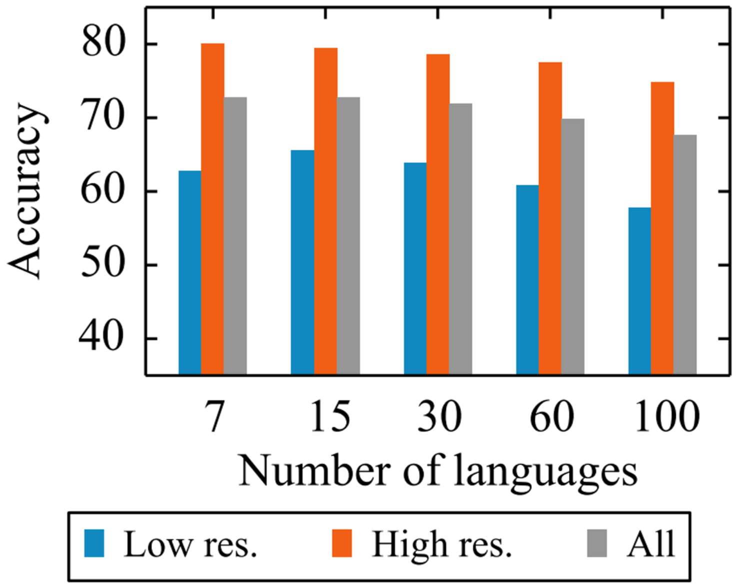
Figures from Conneau et al. [Unsupervised Cross-lingual Representation Learning at Scale, 2020]

The curse of multilinguality

- With fixed capacity as we add more languages, the quality:
  - *When 93 langs. added, acc. for 7 eval langs.: 0.718→0.677*
  - *for high-res. Langs. decrease (capacity dilution)*
  - *for low-res. – first increases (due to cross-ling. transfer), then decreases (due to capacity dilution)*
- Increase capacity with number of languages?

For ablations: base model, pre-training on Wikipedia, fine-tuning on XNLI.

Accuracy on XNLI (zero-shot from en?)
High res.: (English + French) / 2
Low res.: (Swahili + Urdu) / 2
All: + German, Russian, Chinese



Figures from Conneau et al. Unsupervised Cross-lingual Representation Learning at Scale, 2020

XLM-R large

- As BERT/RoBERTa large: 24 layers of Transformer, hidden size 1024
  - 270M/550M in base/large model
  - about half of parameters are subword embeddings
- Vocabulary: 250K subwords shared between 100 languages
  - Sentence Piece with a unigram LM directly on raw text
  - MLM pre-trained with full softmax
- 1.5M updates, bs=8192
- 500 V100 GPUs
  - How many days?
  - Compared to RoBERTa: 2x less GPUs, 3x more updates, same bs => about 1 week?

Figures from Conneau et al. Unsupervised Cross-lingual Representation Learning at Scale, 2020

XLM-R (Summary)

- Pre-training a single MLM on 100 languages:
  - requires high capacity (large vocab., hidden size)
  - requires long training with low-res. langs upsampling
- Fine-tuning multilingual MLM:
  - enables zero-shot cross-lingual transfer
  - but better translate (MT) into all target languages and fine-tune on multilingual train set

Multilingual seq2seq models

- mT5
  - *Pretrained with a standard monolingual T5 denoising objective on mC4 (100 languages)*
  - *Fine-tuned for each task separately*
- mBART
  - *Pretrained with a standard BART denoising objective on 25 languages; later extended to 50 languages*
  - *Language id is specified by the BOS token*
  - *Fine-tuned on translation pairs, achieved SOTA on low- and mid-resource languages*
- M2M100 and related models
  - *An mBART-like transformer trained to translate between 2200 language pairs and 100 languages*
  - *The encoder produces nearly language-agnostic embeddings*

The **denoising objective** is a training goal where the model learns to **reconstruct the original text** from a **corrupted (noisy) version.**

Noise can involve deleting, masking, or shuffling tokens.
🧠 Example:
Input: “<mask> is a large language model”
Target: “ChatGPT is a large language model”

This helps the model learn **deep language understanding** and improves performance on downstream tasks.

https://arxiv.org/abs/2010.11934 ; https://arxiv.org/abs/2001.08210; https://arxiv.org/abs/2010.11125

Multilingual generation

- [XGLM](https://arxiv.org/abs/2112.10668) by Meta
  - *Pretrain a GPT-like model on a balanced corpus of 30 languages*
  - *Probe with few-shot in-context learning*
    - *SOTA in some generation tasks for lower-resourced languages*
    - *Capable of few-shot translation*
- [mGPT](https://arxiv.org/abs/2204.07580) by Sber
  - *Pretrained on 61 languages from Wikipedia and MC4*
  - *High scores in many zero-shot and few-shot tasks*
- Both models can be fine-tuned for specific tasks or used in the zero-/few-shot setting

https://arxiv.org/abs/2112.10668 ; https://arxiv.org/abs/2204.07580

mGPT

mGPT

- mGPT is a multilingual version of GPT-3 trained for **61 languages from 25 language families**, available in 2 versions:
  - *mGPT*$_{1.3B}$,
  - *mGPT*$_{13B}$.

- **23 monolingual versions of mGPT 1.3B** fine-tuned for Georgian and other **Commonwealth of Independent States (CIS) languages** and **under-resourced languages of the small peoples in Russia**.

Motivation

- Develop a **large-scale multilingual autoregressive LM** that inherits the GPT-3's generalization benefits.

- **Increase the linguistic diversity** of multilingual LMs.

- Address languages of the **CIS** and **under-resourced languages of the small peoples in Russia**.

mGPT architecture & training details

- Architecture is based on GPT-3
- 61 languages
- 2 model versions: $mGPT_{1.3B}$ & $mGPT_{13B}$
- 600GB of training texts from Wikipedia & C4
- Data deduplication by Google MT5 guidelines + data filtration
- Unified BBPE tokenizer
- Pretraining with DeepSpeed library & MegatronLM

mGPT languages

- 61 languages
- 12 languages of CIS nations
- 8 under-resourced languages of the small peoples in Russia:
  - *Yakut*
  - *Kalmyk*
  - *Tuvan*
  - *Buryat*
  - *Ossetian*
  - *Chuvash*
  - *Bashkir*
  - *Tatar*

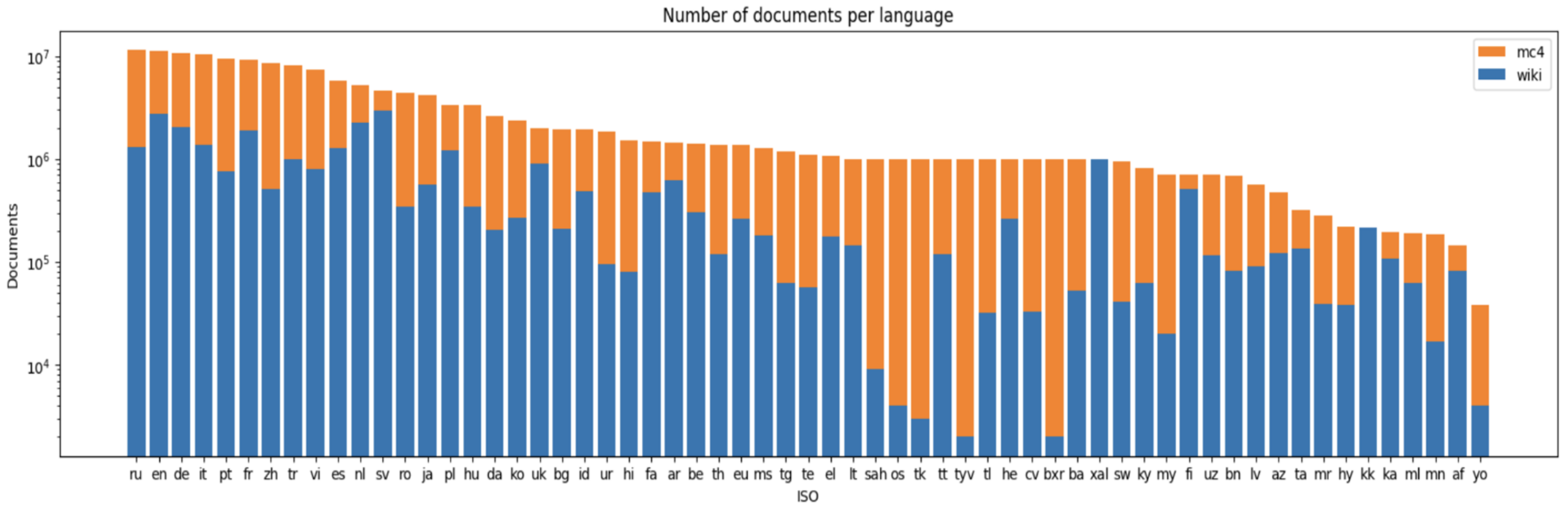


Figure 2: Number of documents for each language in the pretraining corpus on a logarithmic scale.

Tokenization strategies

- DEFAULT: BBPE (Wang et al., 2020);
- CASE: Each uppercase character is replaced with a special token <case> followed by the corresponding lowercase character;
- ARITHMETIC: The CASE strategy combined with representing numbers and arithmetic operations as individual tokens;
- COMBINED: The ARITHMETIC strategy combined with representing punctuation marks and whitespaces as individual tokens;
- CHAR: Character-level tokenization.

Strategy	Tokenization Example
DEFAULT	22, Birds, +, 3, birds, =, 25, birds
CASE	22, <case>, birds, +, 3, birds, ...
ARITHMETIC	2, 2, <case>, birds, , +, , 3, ...
COMBINED	2, 2, <case>, birds, , +, , 3, , ...
CHAR	2, 2, , B, i, r, d, s, , +, , ...

Table 2: Different tokenization strategies applied to the sentence "22 Birds + 3 birds = 25 birds". The resulting tokens are highlighted in the corresponding colors.

Pretrain five strategy-specific versions of mGPT$_{163M}$ on a Wikipedia subset of the pretraining corpus.
The tokenization strategy is

Strategy	Avg. PPL
DEFAULT	**6.94**
CASE	8.13
ARITHMETIC	7.99
COMBINED	8.43
CHAR	9.47

Table 3: The average perplexity results. The best score is put in bold, the second best is underlined.

LM evaluation

- Language modeling performance on the held-out sets for each language.
- Correlation analysis to examine the effect of:
  - *Language script*
  - *Pretraining corpus size*
  - *Model size*

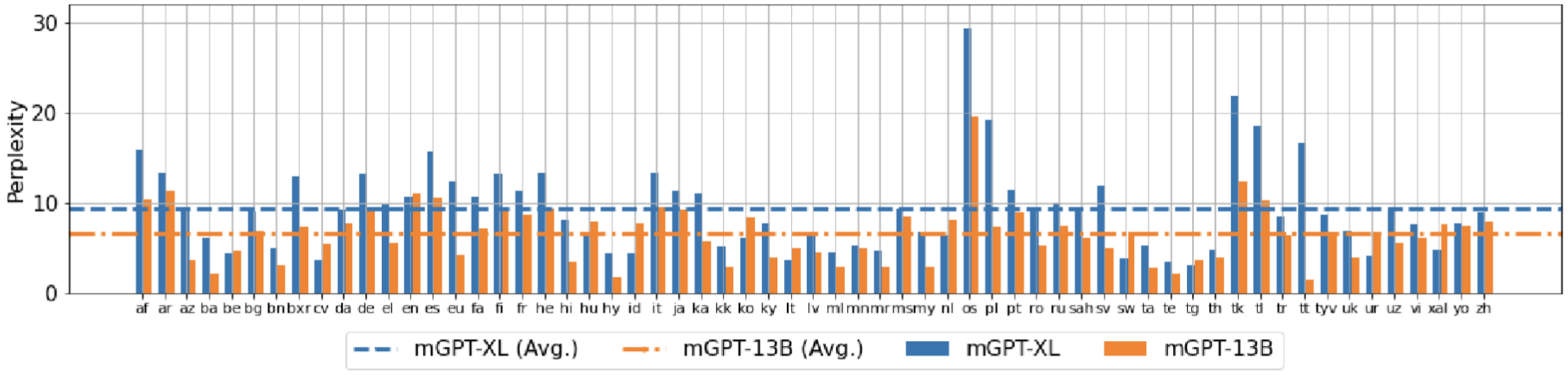


Figure 3: Language-wise perplexity results. Lower is better.

Takeaways:

- $mGPT_{1.3B}$ results have similar distribution as $mGPT_{13B}$ but are consistently higher.
- Non-Latin languages receive lower scores on average, while $mGPT_{13B}$ performs better than $mGPT_{1.3B}$ in this setting.

Criterion	Model	Test	p-value
Language script	$mGPT_{1.3B}$	M-W U test	0.012
	$mGPT_{13B}$		0.000
Pretraining corpus size	$mGPT_{1.3B}$	Pearson	0.137
	$mGPT_{13B}$		0.307
Model size	$mGPT_{1.3B}$	M-W U test	0.0007
	$mGPT_{13B}$		

Table 5: Correlation analysis results.

Downstream evaluation

Extrinsic evaluation of mGPT on **classification** and **sequence labeling** tasks in *zero-/few-shot settings*.

Zero-/few-shot approach based on per-token cross-entropy loss:

- <u>for classification:</u> select the label associated with the prompt that results in the lowest sum of negative log probabilities for its tokens.
- <u>for sequence labeling:</u> at each step select the tag with the lowest sum of losses per token.

Task	**Template**	**Output Candidates**
XNLI	`<s> {sentence 1}`, right? `{label} {sentence 2} </s>`	Yes (Entailment); Also (Neutral) No (Contradiction)
PAWSX	`<s> {sentence 1}`, right? `{label} {sentence 2} </s>`	Yes; No
XWINO	`<s> {sentence start}{candidate} {sentence end} </s>`	✗
XCOPA	`<s> {sentence}` because `{candidate answer} </s>` `<s> {sentence}` so `{candidate answer} </s>`	✗
Hate Speech	`<s> The sentence is {label}. {sentence} </s>`	sexist, racist, offensive, abusive, hateful (Positive) normal, common, ok, usual, acceptable (Negative)
NER	`<s>`lang: `{lang}` \n Tagged sentence: `{sentence with tags}`	I-LOC, I-MISC, I-ORG, I-PER, O
POS	`<s>`lang: `{lang}` \n Tagged sentence: `{sentence with tags}`	ADJ, ADP, ADV, AUX, CCONJ, DET, INTJ, NOUN, NUM, PART, PRON, PROPN, PUNCT, SCONJ, SYM, VERB, X

Table 6: Prompt examples for each downstream task. The examples are in English for illustration purposes.

Downstream evaluation: classification

Takeaways:

- $mGPT_{1.3B}$ is comparable to $XGLM_{1.7B}$ while having fewer weights and covering a larger amount of languages.
- More demonstrations may result in performance degradation.
- Hate speech detection is one of the most challenging tasks.

Model	k-shot	XWINO	PAWSX	XCOPA	XNLI	Hate Speech
$mGPT_{1.3B}$	0	56.2	53.1	55.5	40.6	50.0
	1	57.0	51.3	54.9	36.1	✗
	4	56.8	52.2	54.8	37.4	50.8
	16	54.5	52.2	54.8	37.9	✗
$mGPT_{13B}$	0	59.3	51.5	58.2	42.6	**53.1**
	1	61.0	50.6	57.9	37.5	✗
	4	61.8	51.6	58.3	41.4	51.5
	16	59.2	**55.1**	57.3	33.3	✗
$XGLM_{1.7B}$	0	54.2	50.3	55.5	42.6	50.1
	1	58.0	45.9	56.8	36.4	✗
	4	57.9	45.9	56.2	38.8	49.5
	16	✗	44.2	56.1	36.5	✗
$XGLM_{7.5B}$	0	59.2	50.1	55.5	**44.7**	50.1
	1	63.7	46.4	60.6	36.9	✗
	4	**64.2**	45.3	61.4	40.1	51.8
	16	✗	44.9	**62.5**	40.0	✗

Table 7: Accuracy scores (%) on classification tasks averaged across languages.

Downstream evaluation: sequence labeling

Takeaways:

- $mGPT_{1.3B}$ outperforms $mGPT_{13B}$ on sequence labeling tasks
- mGPT1.3B performance on low-resource languages is comparable with its performance on XGLUE benchmark.

Model	de	en	es	nl	Avg.
Random	1.9	3.1	1.8	1.6	2.1
$mGPT_{1.3B}$	12.2	22.1	12.7	13.1	15.0
$mGPT_{13B}$	5.6	20.9	10.4	6.7	10.9
$M\text{-}BERT_{base}$	69.2	90.6	75.4	77.9	78.2
$XLM\text{-}R_{base}$	70.4	90.9	**75.2**	79.5	79.0
Unicoder	**71.8**	**91.1**	74.4	**81.6**	**79.7**

Table 9: F1-scores for NER by language. The mGPT models are evaluated in the 4-shot setting. The best score is put in bold, the second best is underlined.

Model	XGLUE																			CIS & Low-Resource UD						
	ar	bg	de	el	en	es	fr	hi	it	nl	pl	pt	ru	th	tr	ur	vi	zh	Avg.	be	bxr	hy	kk	sah	tt	uk
Random	6.5	6.5	6.0	5.2	4.4	5.7	5.5	6.7	6.6	6.6	5.9	4.7	6.0	6.4	6.8	1.2	7.0	7.1	5.8	1.3	5.7	5.9	2.6	9.6	8.7	4.8
$mGPT_{1.3B}$	16.5	24.5	30.6	20.9	40.0	24.3	27.0	16.2	25.4	28.8	28.3	24.6	29.4	12.9	30.4	15.0	25.6	19.5	24.4	**21.5**	**28.4**	**14.7**	**22.8**	**19.9**	**21.4**	**22.5**
$mGPT_{13B}$	11.7	21.8	26.8	16.1	36.0	22.2	25.0	12.3	26.5	26.5	24.2	21.8	21.8	9.5	26.8	12.7	21.5	12.5	20.9	10.6	7.7	7.3	9.4	11.8	9.2	10.9
$M\text{-}BERT_{base}$	52.4	85.0	88.7	81.5	95.6	86.8	87.6	58.4	91.3	88.0	81.8	88.3	78.8	43.3	69.2	53.8	54.3	58.3	74.7	✗	✗	✗	✗	✗	✗	✗
$XLM\text{-}R_{base}$	67.3	**88.8**	**92.2**	**88.2**	**96.2**	**89.0**	**89.9**	**74.5**	**92.6**	**88.5**	**85.4**	**89.7**	**86.9**	**57.9**	72.7	**62.1**	55.2	**60.4**	**79.8**	✗	✗	✗	✗	✗	✗	✗
Unicoder	**68.6**	88.5	92.0	88.3	96.1	89.1	89.4	69.9	92.5	88.9	83.6	89.8	86.7	57.6	**75.0**	59.8	**56.3**	60.2	79.6	✗	✗	✗	✗	✗	✗	✗

Table 10: Accuracy scores (%) for XGLUE and Universal Dependencies POS-tagging by language. mGPT models are evaluated in the 4-shot setting. The best score is put in bold, the second best is underlined.

Knowledge probing

Model probing in 23 languages on the mLAMA dataset.

<u>Takeaways:</u>

- Scaling the number of model parameters usually boosts the performance for high-resource.

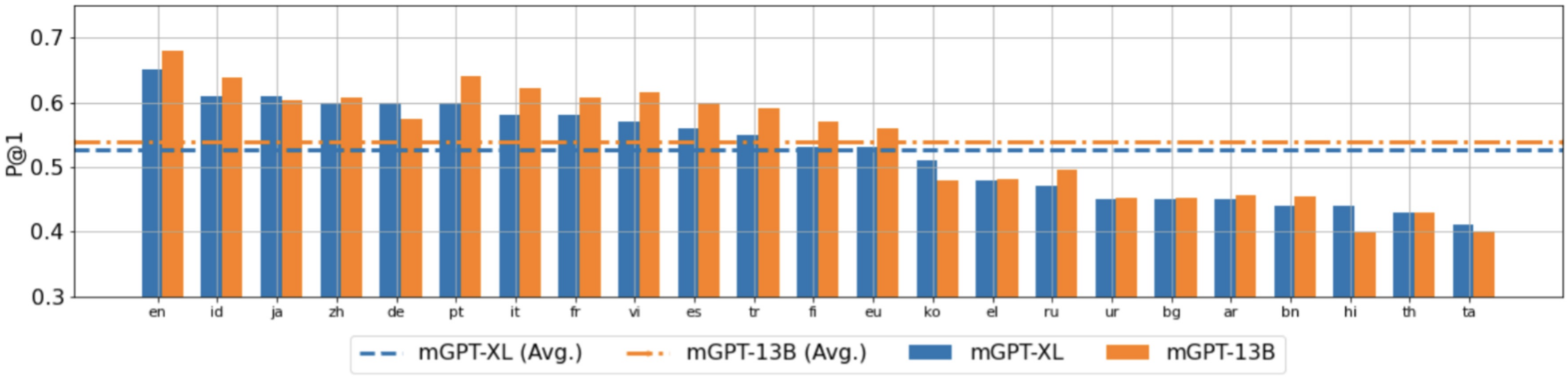
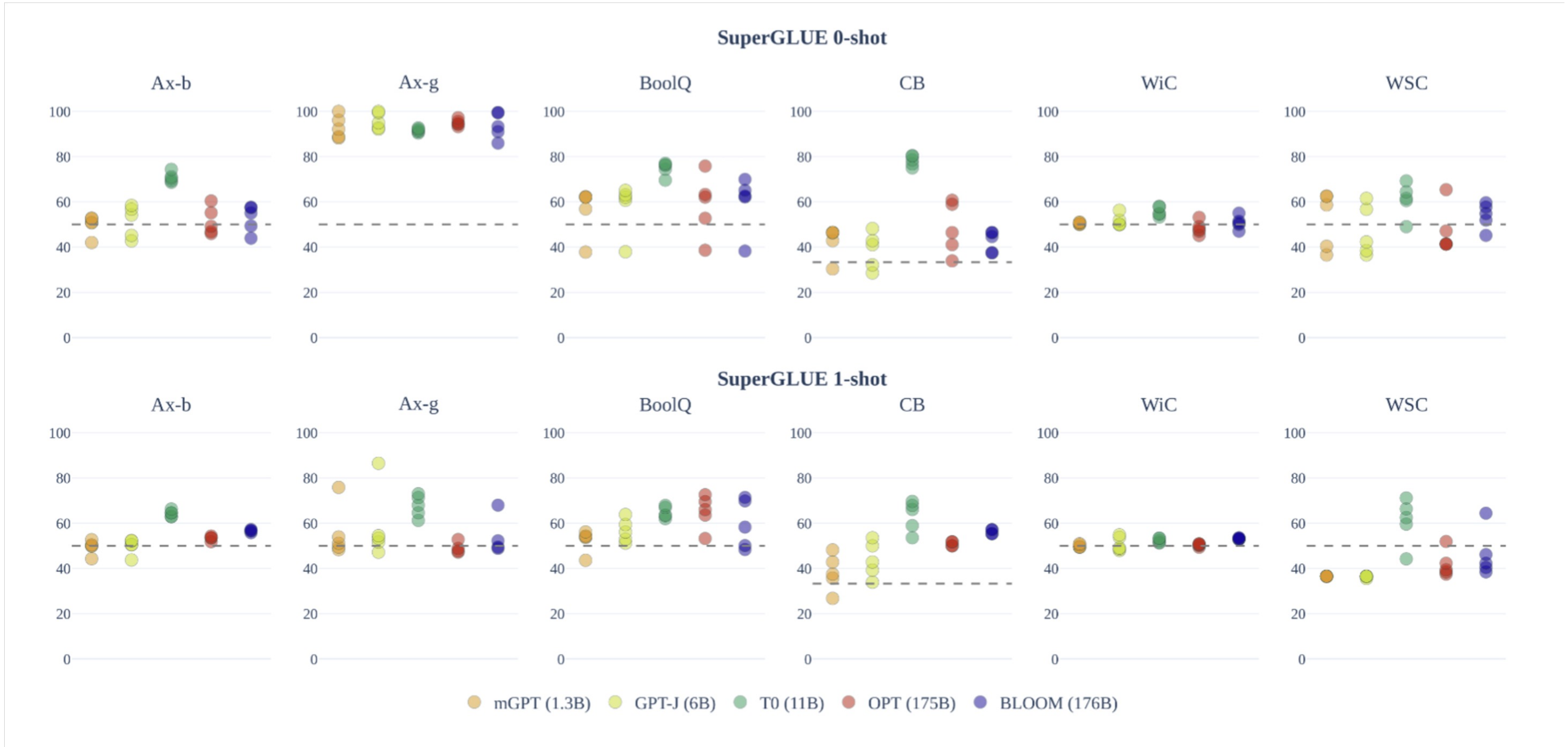


Figure 5: Knowledge probing results for 23 languages. The performance of a random baseline is 0.33.

General language Evaluation

Comparison of BLOOM_{176B}, $\text{mGPT}_{1.3B}$, $\text{OPT}_{175B,}$ GPT-J_{6B} , and T0_{11B} on *SuperGlue benchmark.*

The $\text{mGPT}_{1.3B}$ model has comparable performance despite having fewer weights.



Language Generation

Lexical diversity analysis of $mGPT_{1.3B}$ in five languages: English, French, German, Spanish, and Chinese

- *$Entropy_1$* - the Shannon Entropy over unigrams ,
- *MSTTR* - the mean segmented type-token ratio over segment lengths of 100,
- *$Distinct_1$* - the ratio of distinct unigrams over the total number of unigrams, and the counter of unigrams that appear once in the collection of generated outputs,
- *$Unique_1$* - the counter of unigrams that appear once in the collection of generated outputs

ISO	Avg. length	$Distinct_1$	Vocabulary size	$Unique_1$	$Entropy_1$	TTR	MSTTR
en	39.13 ± 22.61	0.071	387	103	6.175	0.097	0.228
fr	23.53 ± 17.92	0.128	486	181	6.875	0.159	0.346
de	30.85 ± 17.33	0.113	453	159	6.850	0.151	0.340
es	12.71 ± 15.54	0.102	413	124	6.818	0.148	0.315
zh	3.157 ± 2.39	0.492	188	124	7.055	0.525	0.526

Table 11: The results for lexical diversity of generated texts on the GEM story generation task.

Takeaways:

- The diversity metrics scores for Chinese are the highest, while the mean generated text length is the shortest.
- The results for the Indo-European languages are similar (French, German, and Spanish). The metrics are lower for English, with the average text length being longer.

Monolingual fine-tunes

23 monolingual versions of $\mathbf{mGPT_{1.3B}}$ fine-tuned for **Commonwealth of Independent States (CIS) languages** and **under-resourced languages of the small nationalities in Russia**.

General data:

- *mC4*
- *OSCAR*
- *OpenSubtitles*
- *Wiki*
- *Blogs*
- *LibGen*
- *Archive*

Specific text corparas:

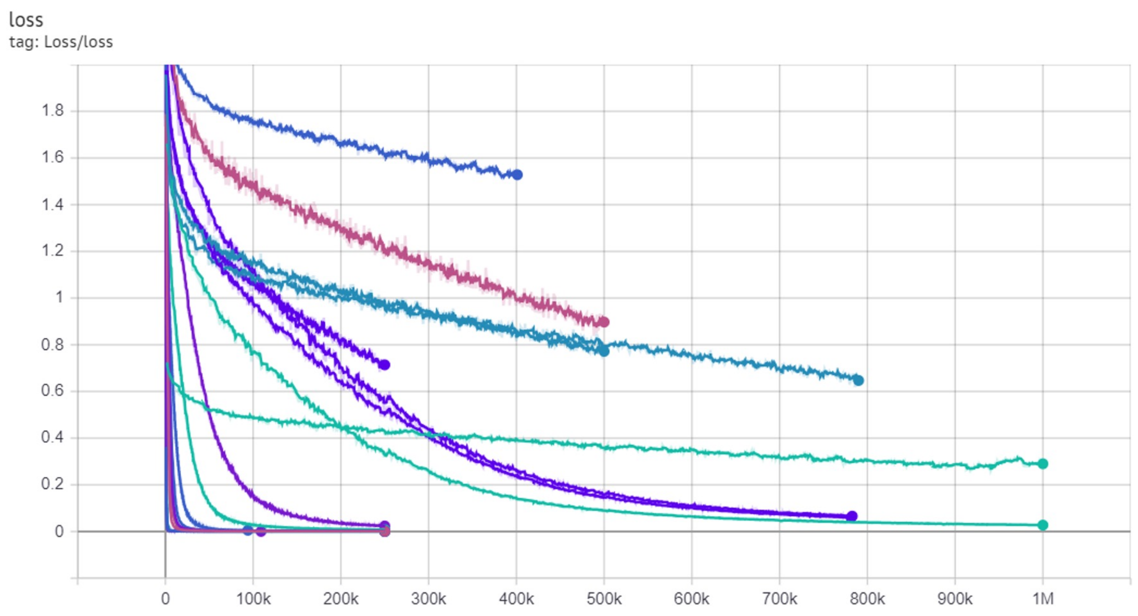
- *EANC*
- *TED talks*
- *minorlangs*
- *Bashkir*
- *XNLI*
- *chuvash_parallel, chv_corpus*
- *oyrad_corpus*
- *Udhr*
- *VOA Corpus*

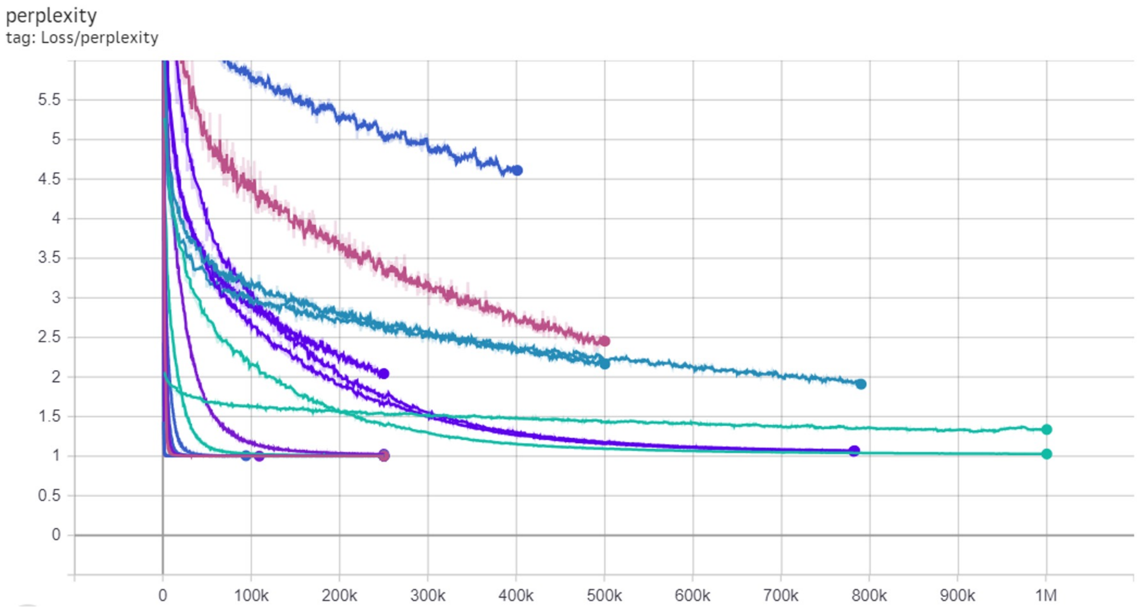
Language	HuggingFace URL	PPL
Armenian	hf.co/ai-forever/mGPT-1.3B-armenian	1.7
Azerbaijan	hf.co/ai-forever/mGPT-1.3B-azerbaijan	5.4
Bashkir	hf.co/ai-forever/mGPT-1.3B-bashkir	7.1
Belorussian	hf.co/ai-forever/mGPT-1.3B-belorussian	27.7
Bulgarian	hf.co/ai-forever/mGPT-1.3B-belorussian	15.2
Buryat	hf.co/ai-forever/mGPT-1.3B-buryat	17.6
Chuvash	hf.co/ai-forever/mGPT-1.3B-chuvash	28.8
Georgian	hf.co/ai-forever/mGPT-1.3B-georgian	16.9
Kalmyk	hf.co/ai-forever/mGPT-1.3B-kalmyk	14.0
Kazakh	hf.co/ai-forever/mGPT-1.3B-kazakh	3.4
Kirgiz	hf.co/ai-forever/mGPT-1.3B-kirgiz	8.2
Mari	hf.co/ai-forever/mGPT-1.3B-mari	21.2
Mongol	hf.co/ai-forever/mGPT-1.3B-mongol	4.4
Ossetian	hf.co/ai-forever/mGPT-1.3B-ossetian	18.7
Persian	hf.co/ai-forever/mGPT-1.3B-persian	33.4
Romanian	hf.co/ai-forever/mGPT-1.3B-romanian	3.4
Tajik	hf.co/ai-forever/mGPT-1.3B-tajik	6.5
Tatar	hf.co/ai-forever/mGPT-1.3B-tatar	3.7
Turkmen	hf.co/ai-forever/mGPT-1.3B-turkmen	28.5
Tuvan	hf.co/ai-forever/mGPT-1.3B-tuvan	40.8
Ukranian	hf.co/ai-forever/mGPT-1.3B-ukranian	7.1
Uzbek	hf.co/ai-forever/mGPT-1.3B-uzbek	6.8
Yakut	hf.co/ai-forever/mGPT-1.3B-yakut	10.6

Table 12: A list of the $mGPT_{1.3B}$ models continuously pretrained on monolingual corpora for 23 languages.

Fine-tuning process

- Initially, models were tuned for 250k steps.
- Checkpoints were saved every 10k steps.
- Additional tuning for some models until they reached the plato.





Comparison with mGPT

mGPT and monolingual mGPT were compared on a test dataset. The quality of the multilingual model MGPT was improved from 47% to 2900% depending on the language

Languages	monolingual mGPT	mGPT	growth
Armenian	loss: 0.55 ppl: 1.73	loss: 1.83 ppl: 6.23	260%
Azerbaijani	loss: 1.68 ppl: 5.37	loss: 3.86 ppl: 47.46	783%
Bashkir	loss: 1.95 ppl: 7.06	loss: 3.56 ppl: 35.01	395%
Belarusian	loss: 3.32 ppl: 27.65	loss: 4.14 ppl: 62.73	126%
Bulgarian	loss: 2.72 ppl: 15.20	loss: 4.45 ppl: 85.70	463%
Buriat	loss: 2.87 ppl: 17.63	loss: 4.19 ppl: 65.70	272%
Chuvash	loss: 3.36 ppl: 28.76	loss: 4.79 ppl: 120.66	319%
Georgian	loss: 2.82 ppl: 16.85	loss: 4.05 ppl: 57.20	239%
Kalmyk	loss: 2.64 ppl: 13.97	loss: 4.30 ppl: 74.12	430%
Kazakh	loss: 1.22 ppl: 3.38	loss: 3.27 ppl: 26.43	680%
Kirgiz	loss: 2.10 ppl: 8.20	loss: 4.23 ppl: 68.44	734%
Mari	loss: 3.05 ppl: 21.19	loss: 5.26 ppl: 193.193	811%
Mongolian	loss: 1.47 ppl: 4.35	loss: 3.32 ppl: 27.69	536%
Ossetian	loss: 2.93 ppl: 18.70	loss: 4.36 ppl: 78.17	318%
Persian	loss: 3.51 ppl: 33.44	loss: 4.45 ppl: 86.05	157%
Romanian	loss: 1.24 ppl: 3.44	loss: 1.63 ppl: 5.08	47%
Tajik	loss: 1.88 ppl: 6.52	loss: 4.09 ppl: 59.88	818%
Tatar	loss: 1.31 ppl: 3.69	loss: 3.17 ppl: 23.84	546%
Turkmen	loss: 3.35 ppl: 28.47	loss: 5.29 ppl: 199.11	600%
Tuvinian	loss: 3.71 ppl: 40.84	loss: 5.10 ppl: 164.40	302%
Ukrainian	loss: 1.96 ppl: 7.11	loss: 4.00 ppl: 54.93	672%
Uzbek	loss: 1.92 ppl: 6.84	loss: 5.33 ppl: 206.85	2924%
Yakut	loss: 2.37 ppl: 10.65	loss: 4.31 ppl: 74.74	611%

Generation examples

GEORGIAN ⇄ ENGLISH

ხელოვნური ინტელექტი, რომელიც მიზნად ისახავდა მეცნიერების, ხელოვნების, ლიტერატურის, მუსიკის, თეატრის, კინოს, თეატრალური ხელოვნების, კინოეკრან

Artificial intelligence aimed at the development of science, art, literature, music, theater, cinema, theatrical art, cinema

MONGOLIAN ⇄ ENGLISH

Хиймэл оюун ухаан нь хүнийг хөгжүүлэхэд чухал үүрэг гүйцэтгэдэг.

Artificial intelligence plays an important role in the development of a person.

ARMENIAN ⇄ ENGLISH

Արհեստական ինտելեկտը մեծ ոլորտ է, որը ներառում է մեքենայական ուսուցում և նյարդային ցանցեր

Artificial intelligence is a large field that includes machine learning and neural networks.

Use in transformers

```python
# Load model directly
from transformers import AutoTokenizer, AutoModelForCausalLM

tokenizer = AutoTokenizer.from_pretrained("ai-forever/mGPT")
model = AutoModelForCausalLM.from_pretrained("ai-forever/mGPT")
```

Limitations

- List of languages is incomplete
- Lack of training data for low resource languages
- Optimal tokenization still remains an open question
- Search for optimal generation parameters

BLOOM

BigScience Large Open-science Open-access Multilingual Language Model

https://arxiv.org/pdf/2211.05100.pdf

BigScience Large Open-science Open-access Multilingual Language Model

- A 176 billion parameter language model;
- Trained on 46 natural languages and 13 programming languages;
- Final run of 117 days (March 11 - July 6) training
- Developed and released by a collaboration of 1000 researchers from 70+ countries and 250+ institutions;
- On the Jean Zay supercomputer in the south of Paris, France;
- Compute grant worth an estimated €3M from French research agencies CNRS and GENCI;
- Presented at the BigScience Workshop in 2022.

https://arxiv.org/pdf/2211.05100.pdf

Model Architecture

General architecture:

- **Type**: Causal transformer (decoder-only, like GPT)
- **Size**: 176B parameters
- **Training corpus**: ROOTS dataset (~1.6TB of multilingual text)
- **Main use**: Text generation, multilingual tasks
- **Tokenization:** Byte-level BPE + Pre-tokenizer (RegExp)
- **Vocabulary Size:** 250K tokens

Other details:

- ALiBi Positional Embeddings
- LayerNorm (PreNorm)

Tokenizer	fr	en	es	zh	hi	ar
Monolingual	1.30	1.15	1.12	1.50	1.07	1.16
BLOOM	1.17 (-11%)	1.15 (+0%)	1.16 (+3%)	1.58 (+5%)	1.18 (+9%)	1.34 (+13%)

Table 2: Fertilities obtained on Universal Dependencies treebanks on languages with existing monolingual tokenizers. The monolingual tokenizers we used were the ones from CamemBERT (Martin et al., 2020), GPT-2 (Radford et al., 2019), `DeepESP/gpt2-spanish`, `bert-base-chinese`, `monsoon-nlp/hindi-bert` and Arabic BERT (Safaya et al., 2020), all available on the HuggingFace Hub.

https://arxiv.org/pdf/2211.05100.pdf

ALiBi Positional Embeddings

ALiBi (Attention with Linear Biases) is a positional encoding method that:

- **Does not add explicit positional embeddings**.
- Instead, it adds a **linear bias** directly to the attention scores, depending on the **distance** between tokens.

https://arxiv.org/pdf/2211.05100.pdf

How it works?

In standard attention, the attention scores are computed as:

$$\text{Scores} = \frac{QK^T}{\sqrt{d_k}}$$

In ALiBi, this becomes:

$$\text{Scores} = \frac{QK^T}{\sqrt{d_k}} + \text{Bias}$$

Where the **bias** is:

$$\text{Bias}_{i,j} = -s_h \cdot |i - j|$$

- s_h is a **slope** specific to each attention head,
- $|i - j|$ is the distance between query and key positions.

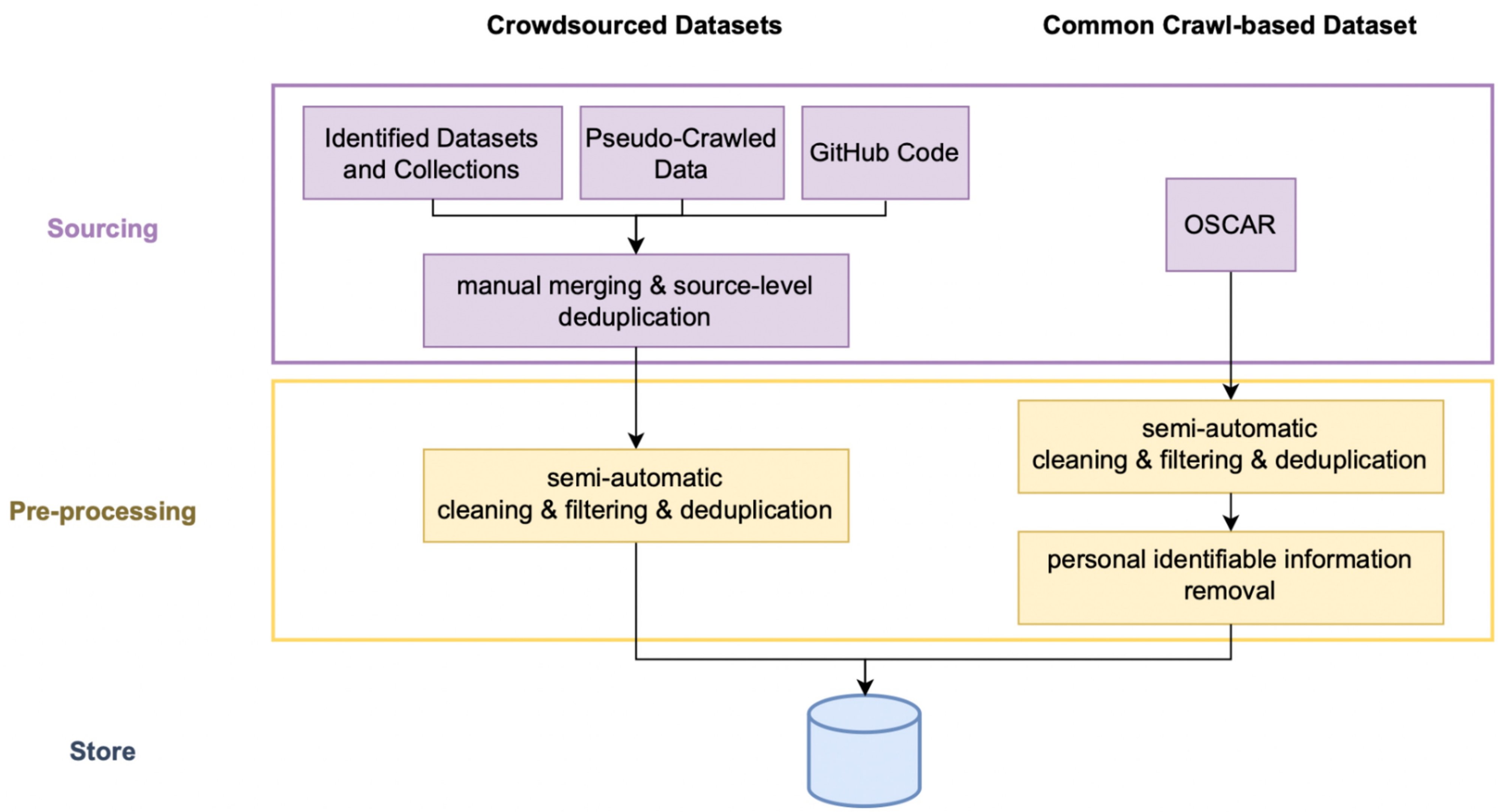
This makes attention **favor closer tokens** (since farther tokens get more negative bias), while **still allowing long-range attention** if necessary.

ALiBi Positional Embeddings

<u>Why it's useful:</u>

- **No position embeddings needed** => simplifies the model.
- **Works for any input length** => no fixed maximum sequence length.
- **Generalizes better** to longer sequences than learned or sinusoidal encodings.
- **Lighter and faster** at inference and training.

Dataset



https://arxiv.org/pdf/2211.05100.pdf

Language	ISO-639-3	catalog-ref	Genus	Family	Macroarea	Size in Bytes
Akan	aka	ak	Kwa	Niger-Congo	Africa	70,1554
Arabic	arb	ar	Semitic	Afro-Asiatic	Eurasia	74,854,900,600
Assamese	asm	as	Indic	Indo-European	Eurasia	291,522,098
Bambara	bam	bm	Western Mande	Mande	Africa	391,747
Basque	eus	eu	Basque	Basque	Eurasia	2,360,470,848
Bengali	ben	bn	Indic	Indo-European	Eurasia	18,606,823,104
Catalan	cat	ca	Romance	Indo-European	Eurasia	17,792,493,289
Chichewa	nya	ny	Bantoid	Niger-Congo	Africa	1,187,405
chiShona	sna	sn	Bantoid	Niger-Congo	Africa	6,638,639
Chitumbuka	tum	tum	Bantoid	Niger-Congo	Africa	170,360
English	eng	en	Germanic	Indo-European	Eurasia	484,953,009,124
Fon	fon	fon	Kwa	Niger-Congo	Africa	2,478,546
French	fra	fr	Romance	Indo-European	Eurasia	208,242,620,434
Gujarati	guj	gu	Indic	Indo-European	Eurasia	1,199,986,460
Hindi	hin	hi	Indic	Indo-European	Eurasia	24,622,119,985
Igbo	ibo	ig	Igboid	Niger-Congo	Africa	14078,521
Indonesian	ind	id	Malayo-Sumbawan	Austronesian	Papunesia	19,972,325,222
isiXhosa	xho	xh	Bantoid	Niger-Congo	Africa	14,304,074
isiZulu	zul	zu	Bantoid	Niger-Congo	Africa	8,511,561
Kannada	kan	kn	Southern Dravidian	Dravidian	Eurasia	2,098,453,560

https://arxiv.org/pdf/2211.05100.pdf

Language	ISO-639-3	catalog-ref	Genus	Family	Macroarea	Size in Bytes
Kinyarwanda	kin	rw	Bantoid	Niger-Congo	Africa	40,428,299
Kirundi	run	rn	Bantoid	Niger-Congo	Africa	3,272,550
Lingala	lin	ln	Bantoid	Niger-Congo	Africa	1,650,804
Luganda	lug	lg	Bantoid	Niger-Congo	Africa	4,568,367
Malayalam	mal	ml	Southern Dravidian	Dravidian	Eurasia	3,662,571,498
Marathi	mar	mr	Indic	Indo-European	Eurasia	1,775,483,122
Nepali	nep	ne	Indic	Indo-European	Eurasia	2,551,307,393
Northern Sotho	nso	nso	Bantoid	Niger-Congo	Africa	1,764,506
Odia	ori	or	Indic	Indo-European	Eurasia	1,157,100,133
Portuguese	por	pt	Romance	Indo-European	Eurasia	79,277,543,375
Punjabi	pan	pa	Indic	Indo-European	Eurasia	1,572,109,752
Sesotho	sot	st	Bantoid	Niger-Congo	Africa	751,034
Setswana	tsn	tn	Bantoid	Niger-Congo	Africa	1,502,200
Simplified Chinese	—	zhs	Chinese	Sino-Tibetan	Eurasia	261,019,433,892
Spanish	spa	es	Romance	Indo-European	Eurasia	175,098,365,045
Swahili	swh	sw	Bantoid	Niger-Congo	Africa	236,482,543
Tamil	tam	ta	Southern Dravidian	Dravidian	Eurasia	7,989,206,220
Telugu	tel	te	South-Central Dravidian	Dravidian	Eurasia	2993407,159
Traditional Chinese	—	zht	Chinese	Sino-Tibetan	Eurasia	762,489,150
Twi	twi	tw	Kwa	Niger-Congo	Africa	1,265,041

https://arxiv.org/pdf/2211.05100.pdf

Language	ISO-639-3	catalog-ref	Genus	Family	Macroarea	Size in Bytes
Urdu	urd	ur	Indic	Indo-European	Eurasia	2,781,329,959
Vietnamese	vie	vi	Viet-Muong	Austro-Asiatic	Eurasia	43,709,279,959
Wolof	wol	wo	Wolof	Niger-Congo	Africa	3,606,973
Xitsonga	tso	ts	Bantoid	Niger-Congo	Africa	707,634
Yoruba	yor	yo	Defoid	Niger-Congo	Africa	89,695,835
Programming Languages	—	—	—	—		174,700,245,772

https://arxiv.org/pdf/2211.05100.pdf

Data Sources

Language Choices

- started with 8 languages, expanded Swahili, Hindi and Urdu;
- groups of 3 fluent in an additional language could add it

Source Selection

- “BigScience Catalogue”
- Arabic-focused Masader repositor
- 252 sources with at least 21 sources per language category
- Pseudocrawl for Spanish, Chinese, French, and English

GitHub Code (Google’s BigQuery)

OSCAR (38% of the corpus)

https://arxiv.org/pdf/2211.05100.pdf

Engineering

- Trained on Jean Zay, a French government-funded supercomputer owned by GENCI and operated at IDRIS.
- 3.5 months to complete and consumed 1,082,990 compute hours.
- Conducted on 48 nodes, each having 8 NVIDIA A100 80GB GPUs (a total of 384 GPUs).
- A reserve of 4 spare nodes with 2x AMD EPYC 7543 32-Core CPUs and 512 GB of RAM.
- Trained on Megatron-DeepSpeed framework.
- *bfloat16* mixed precision, which proved to solve the instability problem.

https://arxiv.org/pdf/2211.05100.pdf

SuperGLUE

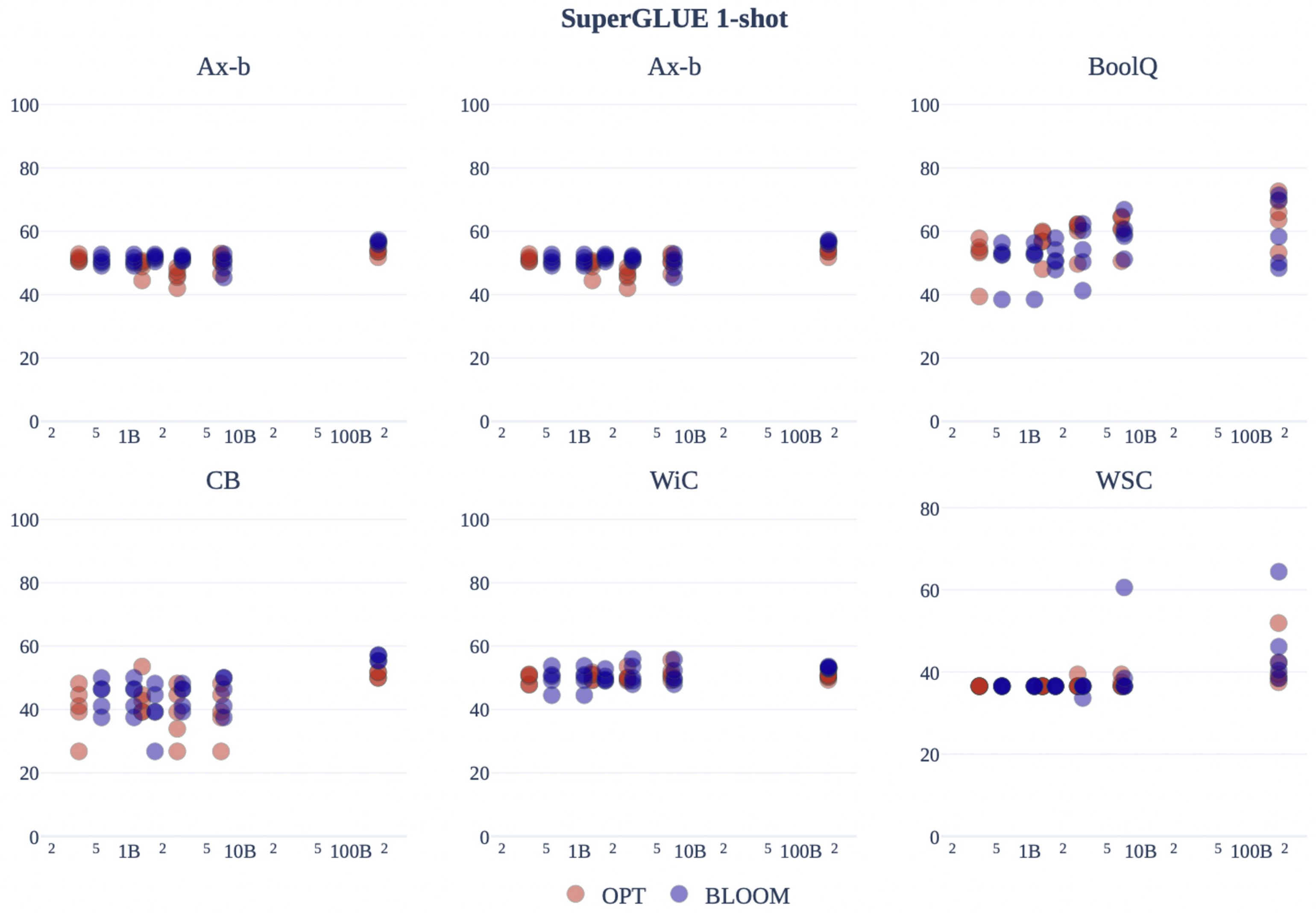


Figure 8: Comparison of the scaling of BLOOM versus OPT on each SuperGLUE one-shot task. Each point represents the average accuracy of a model within the BLOOM or OPT family of models on one of the five task prompts. The number of parameters on the x-axis is presented in log-scale.

https://arxiv.org/pdf/2211.05100.pdf

Machine Translation (1-shot)

Src↓	Trg→	eng	ben	hin	swh	yor
eng	M2M	–	23.04	28.15	29.65	2.17
	BLOOM	–	25.52	27.57	21.7	2.8
ben	M2M	22.86	–	21.76	14.88	0.54
	BLOOM	30.23	–	16.4	–	–
hin	M2M	27.89	21.77	–	16.8	0.61
	BLOOM	35.40	23.0	–	–	–
swh	M2M	30.43	16.43	19.19	–	1.29
	BLOOOM	37.9	–	–	–	1.43
yor	M2M	4.18	1.27	1.94	1.93	–
	BLOOM	3.8	–	–	0.84	–

(a) Low-resource languages

Src↓	Trg→	cat	spa	fre	por
cat	M2M	–	25.17	35.08	35.15
	BLOOM	–	29.12	34.89	36.11
spa	M2M	23.12	–	29.33	28.1
	BLOOM	31.82	–	24.48	28.0
glg	M2M	30.07	27.65	37.06	34.81
	BLOOM	38.21	27.24	36.21	34.59
fre	M2M	28.74	25.6	–	37.84
	BLOOM	38.13	27.40	–	39.60
por	M2M	30.68	25.88	40.17	–
	BLOOM	40.02	28.1	40.55	–

(b) Romance languages

https://arxiv.org/pdf/2211.05100.pdf

Machine Translation (1-shot)

Src ↓	Trg →	eng	fre	hin	ind	vie
eng	M2M	–	41.99	28.15	37.26	35.1
	BLOOM	–	44.4	27.57	38.75	28.83
fre	M2M	37.17	–	22.91	29.14	30.26
	BLOOM	45.11	–	17.04	29.50	31.66
hin	M2M	27.89	25.88	–	21.03	23.85
	BLOOM	35.40	27.83	–	–	–
ind	M2M	33.74	30.81	22.18	–	31.4
	BLOOM	44.59	29.75	–	–	–
vie	M2M	29.51	28.52	20.35	27.1	–
	BLOOM	38.77	28.57	–	–	–

(d) High→mid-resource language pairs.

Src ↓	Trg →	ara	fre	eng	chi	spa
ara	M2M	–	25.7	25.5	13.1	16.74
	XGLM	–	17.9	27.7	–	–
	AlexaTM	–	35.5	41.8	–	23.2
	BLOOM	–	33.26	40.59	18.88	23.33
fre	M2M	15.4	–	37.2	17.61	25.6
	XGLM	5.9	–	40.4	–	–
	AlexaTM	24.7	–	47.1	–	26.3
	BLOOM	23.30	–	45.11	22.8	27.4
eng	M2M	17.9	42.0	–	19.33	25.6
	XGLM	11.5	36.0	–	–	–
	AlexaTM	32.0	50.7	–	–	31.0
	BLOOM	28.54	44.4	–	27.29	30.1
chi	M2M	11.55	24.32	20.91	–	15.92
	XGLM	–	–	–	–	–
	AlexaTM	–	–	–	–	–
	BLOOM	15.58	25.9	30.60	–	20.78
spa	M2M	12.1	29.3	25.1	14.86	–
	XGLM	–	–	–	–	–
	AlexaTM	20.8	33.4	34.6	??	–
	BLOOM	18.69	24.48	33.63	20.06	–

(c) High-resource language pairs.

https://arxiv.org/pdf/2211.05100.pdf

Summarization (1-shot)

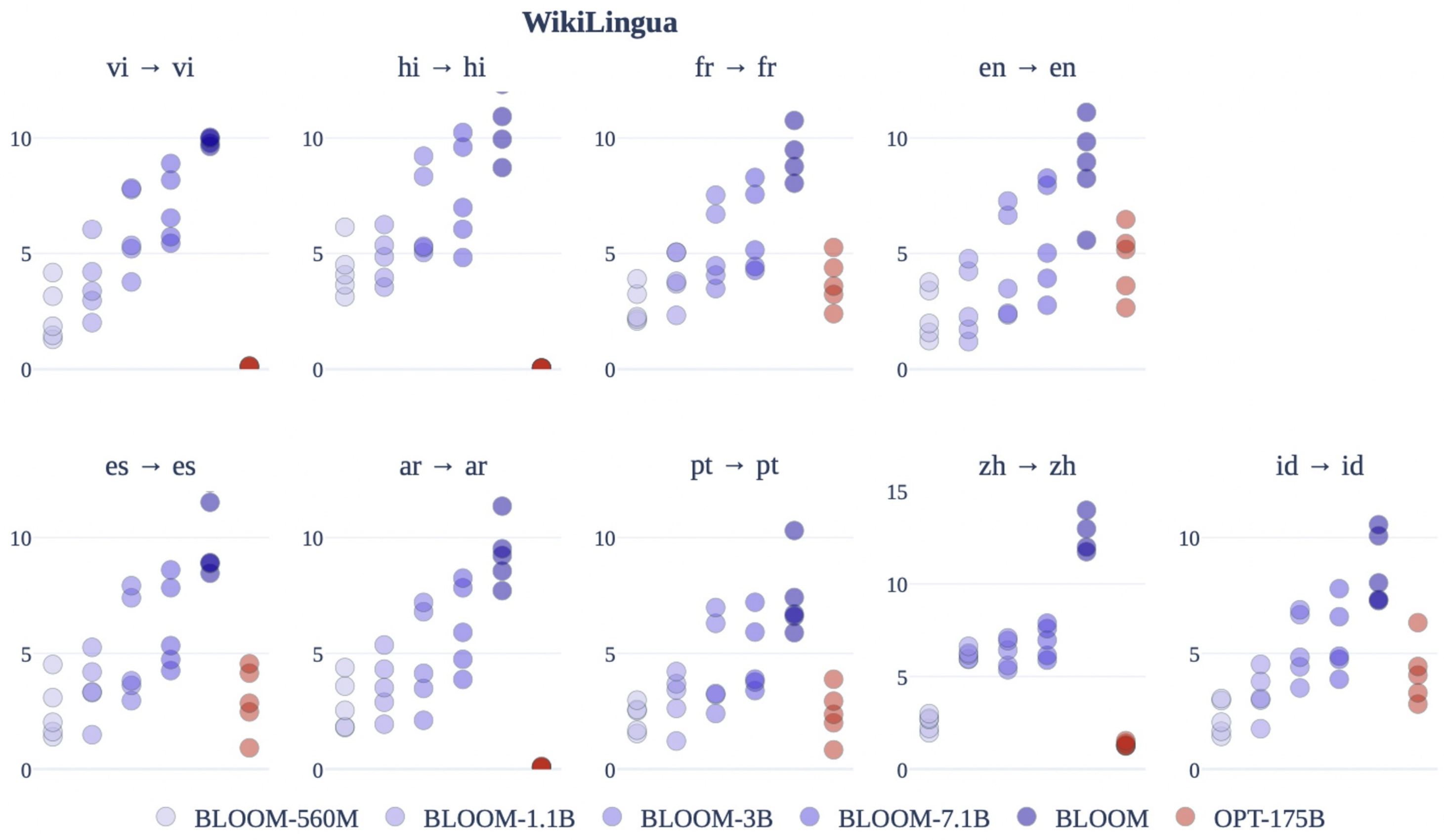


Figure 9: WikiLingua One-shot Results. Each plot represents a different language with per-prompt ROUGE-2 F-measure scores.

https://arxiv.org/pdf/2211.05100.pdf

BLOOM+1

- Add more languages via:
    - Continued Pretraining,
    - MAD-X adapter,
    - (IA)3 technique.

Language	Language Family	Word Order	Script	Space-Separated	Seen Script
German	Indo-European (Germanic)	SVO	Latin	✓	✓
Bulgarian	Indo-European (Slavic)	SVO	Cyrillic	✓	✗
Russian	Indo-European (Slavic)	SVO	Cyrillic	✓	✗
Greek	Indo-European (Hellenic)	SVO	Greek	✓	✗
Turkish	Turkic	SOV	Latin	✓	✓
Korean	Koreanic	SOV	Hangul	✓	✗
Thai	Tai–Kadai	SVO	Thai	✗	✗
Guarani	Tupian	SVO	Latin	✓	✓

Table 1: Information about the unseen languages used in our experiments.

- Need around 100 million tokens of the new language for effective language adaptation.

- When model sizes increases beyond 3 billion parameters, adapter-based language adaptation methods outperform continued pretraining.

https://arxiv.org/pdf/2212.09535v1.pdf

BLOOM+1

- Add more languages via:
    - Continued Pretraining,
    - MAD-X adapter,
    - (IA)3 technique.

Language	Language Family	Word Order	Script	Space-Separated	Seen Script
German	Indo-European (Germanic)	SVO	Latin	✓	✓
Bulgarian	Indo-European (Slavic)	SVO	Cyrillic	✓	✗
Russian	Indo-European (Slavic)	SVO	Cyrillic	✓	✗
Greek	Indo-European (Hellenic)	SVO	Greek	✓	✗
Turkish	Turkic	SOV	Latin	✓	✓
Korean	Koreanic	SOV	Hangul	✓	✗
Thai	Tai–Kadai	SVO	Thai	✗	✗
Guarani	Tupian	SVO	Latin	✓	✓

Table 1: Information about the unseen languages used in our experiments.

- Need around 100 million tokens of the new language for effective language adaptation.

- When model sizes increases beyond 3 billion parameters, adapter-based language adaptation methods outperform continued pretraining.

https://arxiv.org/pdf/2212.09535v1.pdf

Recap on adapters

Adapters are small, trainable modules inserted into a **frozen pre-trained model** (like BLOOM) to enable **efficient fine-tuning** on new tasks or languages.

Instead of updating all model parameters, adapters allow you to:

- Keep the main model frozen.
- Train only the small adapter layers.
- Save memory and computation.

Each adapter is a small bottleneck module inserted **after the feed-forward network (FFN)** and **after the attention sublayer**, like this:

$$\mathrm{Adapter}(x) = x + W_{\mathrm{up}} \cdot \mathrm{ReLU}(W_{\mathrm{down}} \cdot x)$$

- $W_{\mathrm{down}} \in \mathbb{R}^{r \times d}$: **down-projection matrix**
- $W_{\mathrm{up}} \in \mathbb{R}^{d \times r}$: **up-projection matrix**
- $r \ll d$: the **bottleneck dimension**
  (typically a bottleneck of 8, 16, or 64)

https://arxiv.org/pdf/2212.09535v1.pdf

Adapting BERT to new languages

Adapting BERTs to new languages

- *Simplest:* **fine-tune the model** on the target language

https://aclanthology.org/W19-3712; https://arxiv.org/abs/1905.07213; https://aclanthology.org/2022.acl-short.68/

Adapting BERTs to new languages

- *Simplest:* **fine-tune the model** on the target language
- *Vocabulary adaptation:* more efficient, but more complex
  - *Remove unused tokens from the vocabulary (based on a target-lang corpus)*
  - *Add new tokens (e.g. by adding producing some BPE merges)*
  - *Initialize new embeddings using average embeddings of their constituents or source-language tokens aligned with them*
  - *Fine-tune the model on the target-lang (with e.g. MLM loss)*
    - *To speed it up, only embeddings can be fine-tuned (at least, for the 1st epoch)*

https://aclanthology.org/W19-3712; https://arxiv.org/abs/1905.07213; https://aclanthology.org/2022.acl-short.68/

Adapting BERTs to new languages

- *Simplest:* **fine-tune the model** on the target language
- *Vocabulary adaptation:* more efficient, but more complex
  - *Remove unused tokens from the vocabulary (based on a target-lang corpus)*
  - *Add new tokens (e.g. by adding producing some BPE merges)*
  - *Initialize new embeddings using average embeddings of their constituents or source-language tokens aligned with them*
  - *Fine-tune the model on the target-lang (with e.g. MLM loss)*
    - *To speed it up, only embeddings can be fine-tuned (at least, for the 1st epoch)*
- *Training from scratch* (which is more expensive)

https://aclanthology.org/W19-3712; https://arxiv.org/abs/1905.07213; https://aclanthology.org/2022.acl-short.68/

Tips for multilingual classification

- Augmentation with translated data helps
- Domain and task adaptation usually helps
- Multilingual training usually helps
- Zero-shot transfer works OK, but worse than

Model	Data	DE	FR	JA	ES
multi-target	target	94.1	93.8	91.1	78.1
multi-all	all	93.8	94.3	91.4	77.7
zero-shot	EN	92.7	92.6	88.5	72.1

			CLS					HatEval				
Model	**Adapt.**	**Aug.**	**EN**	**DE**	**FR**	**JA**	**AVG**	**EN**	**EN†**	**ES**	**AVG**	**AVG†**
mono-target												
RoBERTa (EN) BERT (OTHERS)	×	×	$94.7_{0.4}$	$90.9_{0.6}$	$95.2_{0.0}$	$88.7_{0.3}$	92.4	$44.4_{5.3}$	$58.5_{6.2}$	$75.6_{0.6}$	60.0	67.1
		✓	$\mathbf{95.3}_{0.3}$	$92.0_{0.2}$	$95.6_{0.3}$	$89.3_{0.02}$	93.0	$46.1_{2.6}$	$60.6_{3.2}$	$76.0_{1.7}$	61.0	68.3
	TAPT	×	$94.9_{0.1}$	$91.6_{0.1}$	$95.4_{0.1}$	$89.3_{0.3}$	92.8	$45.4_{1.9}$	$59.9_{2.7}$	$76.1_{1.1}$	60.8	68.0
		✓	$95.0_{0.4}$	$92.3_{0.4}$	$95.8_{0.2}$	$89.7_{0.4}$	93.2	$44.7_{1.5}$	$59.2_{1.7}$	$76.9_{1.4}$	60.8	68.0
	TAPT+ DAPT	×	$94.9_{0.4}$	$91.8_{0.2}$	$95.5_{0.3}$	$89.5_{0.2}$	92.9	$48.0_{1.5}$	$63.1_{2.6}$	$76.3_{1.1}$	62.2	69.7
		✓	$\mathbf{95.3}_{0.1}$	$93.0_{0.8}$	$\mathbf{95.9}_{0.1}$	$89.9_{0.4}$	**93.5**	$46.0_{4.3}$	$60.2_{4.4}$	$76.9_{0.6}$	61.4	68.5
multi-target												
XLM-RoBERTa	×	×	$92.5_{0.4}$	$93.0_{0.2}$	$92.5_{0.3}$	$90.4_{0.5}$	92.1	$47.2_{2.0}$	$61.4_{1.9}$	$74.8_{0.5}$	61.0	68.1
		✓	$93.3_{0.1}$	$94.0_{0.2}$	$93.8_{0.2}$	$90.3_{0.3}$	92.8	$45.6_{1.6}$	$59.3_{2.5}$	$77.0_{1.1}$	61.3	68.1
	TAPT	×	$92.7_{0.5}$	$93.5_{0.5}$	$93.9_{0.3}$	$90.3_{0.1}$	92.6	$47.0_{2.7}$	$62.4_{3.3}$	$76.1_{1.4}$	61.6	69.2
		✓	$93.4_{0.6}$	$94.0_{0.3}$	$93.8_{0.5}$	$90.5_{0.4}$	92.9	$47.9_{1.3}$	$63.5_{1.5}$	$77.9_{0.9}$	62.9	70.7
	TAPT+ DAPT	×	$93.1_{0.6}$	$93.0_{0.5}$	$93.6_{0.1}$	$90.8_{0.3}$	92.6	$49.9_{2.5}$	$65.6_{2.4}$	$76.5_{1.0}$	63.2	71.0
		✓	$94.0_{0.3}$	$\mathbf{94.1}_{0.4}$	$93.8_{0.3}$	$91.1_{0.4}$	93.2	$46.6_{2.1}$	$61.7_{2.5}$	$\mathbf{78.1}_{0.8}$	62.3	69.9
multi-all												
XLM-RoBERTa	×	×	$92.4_{0.3}$	$92.6_{0.4}$	$93.3_{0.4}$	$90.4_{0.4}$	92.2	$48.4_{3.5}$	$63.1_{4.5}$	$77.5_{0.4}$	62.9	70.3
		✓	$93.4_{0.3}$	$93.3_{0.2}$	$94.0_{0.2}$	$90.4_{0.5}$	92.8	$49.8_{3.5}$	$66.0_{4.6}$	$77.8_{0.9}$	63.8	71.9
	TAPT	×	$92.5_{0.4}$	$93.0_{0.3}$	$93.9_{0.3}$	$90.9_{0.3}$	92.6	$48.4_{2.7}$	$64.2_{3.5}$	$77.4_{0.9}$	62.9	70.8
		✓	$93.5_{0.4}$	$93.4_{0.5}$	$94.1_{0.2}$	$91.1_{0.2}$	93.0	$50.0_{2.2}$	$66.5_{2.6}$	$77.8_{0.6}$	63.9	72.2
	TAPT+ DAPT	×	$92.7_{0.3}$	$93.3_{0.2}$	$94.0_{0.3}$	$91.2_{0.3}$	92.8	$47.1_{3.9}$	$62.7_{5.3}$	$77.4_{1.0}$	62.3	70.1
		✓	$93.5_{0.3}$	$93.8_{0.2}$	$94.3_{0.3}$	$\mathbf{91.4}_{0.2}$	93.3	$\mathbf{50.7}_{1.1}$	$\mathbf{67.4}_{1.4}$	$77.7_{0.7}$	**64.2**	**72.6**

Practical Transformer-based Multilingual Text Classification, Wang and Banco, 2021: https://aclanthology.org/2021.naacl-industry.16

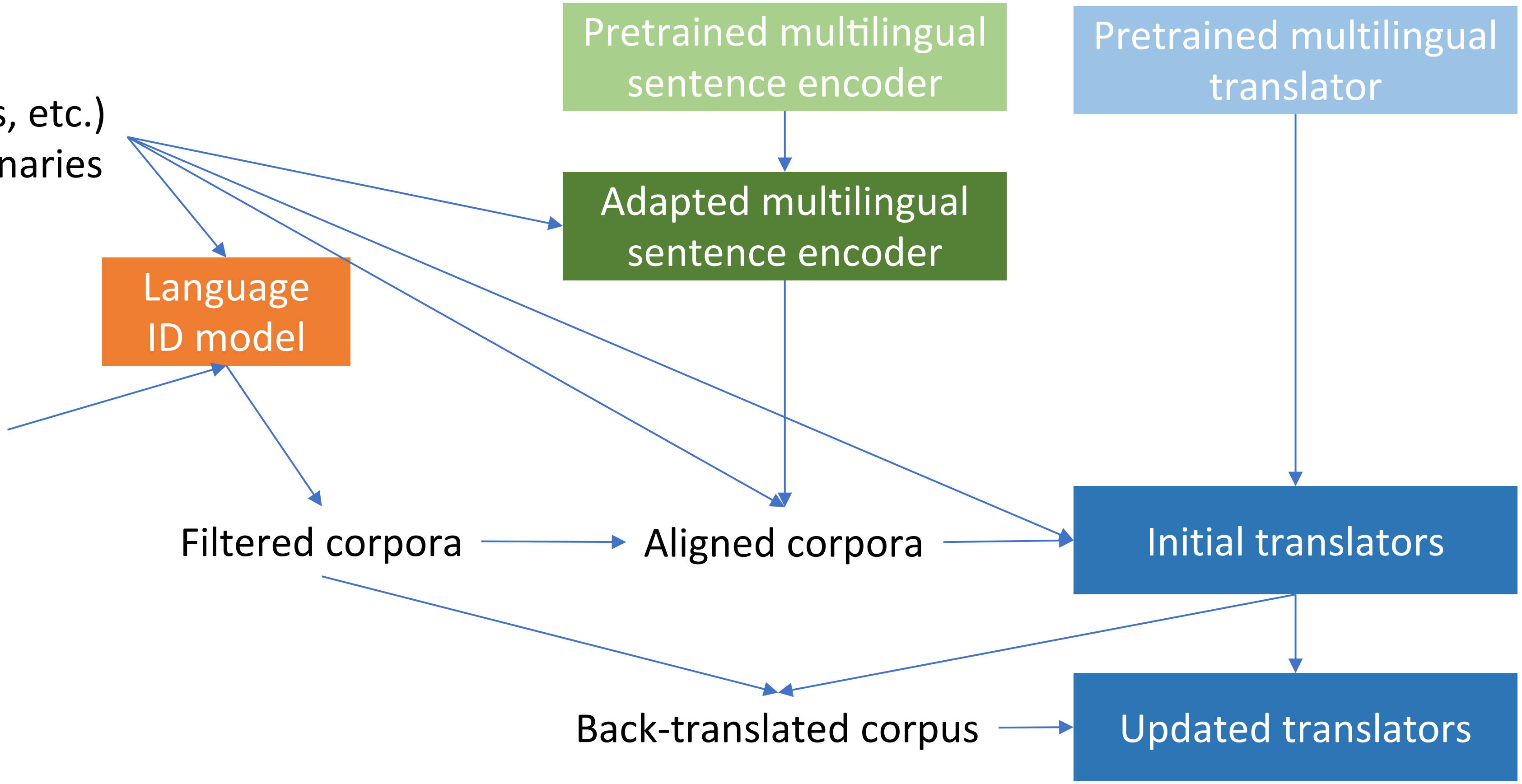
How to bootstrap NLP for a new language?

Typical initial resources:

- Small parallel data (bible, laws, etc.)
- Word- and phrase-level dictionaries
- Wikipedia

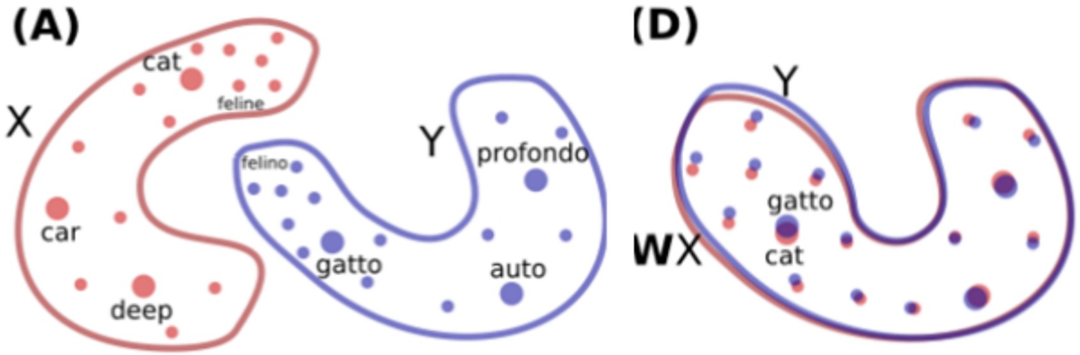
Dirty corpora

- Wikipedia in other languages
- Mixed-language web crawl
- Unaligned parallel literature



Unsupervised word translation

- Hypothesis: Word embedding spaces in two languages are isomorphic
  - One embedding space can be linearly transformed into another
  - Give monolingual embeddings X and Y, learn a (orthogonal) matrix, such that, WX = Y
- Use adversarial learning to learn W:
  - If WX and Y are perfectly aligned, a discriminator shouldn't be able to tell
  - Discriminator: Predict whether an embedding is from Y or the transformed space WX.
  - Train W to confuse the discriminator



After aligning words, a sentence translation model can be trained:

- Pretrain with monolingual denoising
- Finetune with back-translation

http://phontron.com/class/multiling2022/assets/slides/multiling-10-unsupervisedmt.pdf

What is next?

What about LLMs?

- *New horizons with Large Language Modeling*

GPT-4 is OpenAI's most advanced system, producing safer and more useful responses

What about LLMs?

Officially LLaMA 2 models *(=> all models based on it e. g. Mistral*) are intended for the English Language.

Purpose

Llama 2 good for both commercial and academic purposes, specifically targeting English language applications. While tuned models are crafted to function as

Limitations

- Activities that breach legal and regulatory standards, including international trade laws, are discouraged.
- Applications in languages other than English.

In fact, LLaMA models *(=> models based on it e. g. Mistral*) know other languages pretty well.

What about LLMs?

Officially **LLaMA 3.2** models *”speaks” only 8 languages.*In fact, LLaMA is excellent in *Russian, for example*.

Llama 4 officially excels at understanding **12 languages.**

Qwen 2.5 officially supports more than 29 languages, including *Chinese, English, French, Spanish, Portuguese, German, Italian, Russian, Japanese, Korean, Vietnamese, Thai, Arabic, and others.*

DeepSeek R1 officially supports **72 languages**, including: *Chinese, English, Russian, French, Spanish, Portuguese, German, Italian, Japanese, Korean, Arabic, Vietnamese, Thai, Hindi, Indonesian, Turkish, Dutch, Polish, Romanian, Ukrainian, Hebrew, Greek, Czech, Swedish, Finnish, Hungarian, Danish, Norwegian, Malay, Filipino, Bulgarian, Slovak, Serbian, Croatian, Lithuanian, Latvian, Estonian, Slovenian, Bengali, Tamil, Telugu, Kannada, Malayalam, Urdu, Punjabi, Gujarati, Marathi, Sinhala, Nepali, Burmese, Khmer, Lao, Mongolian, Farsi, Pashto, Kurdish, Uzbek, Kazakh, Azerbaijani, Armenian, Georgian, Swahili, Zulu, Xhosa, Hausa, Yoruba, Amharic, Somali, Maori, Haitian Creole, Icelandic, Basque, Galician.*

The full list of languages in text-only mode is:

- English
- German
- French
- Italian
- Portuguese
- Hindi
- Spanish
- Thai

Multilingual Writing

Llama 4 was also pre-trained and fine-tuned for unrivaled text understanding across 12 languages, supporting global development and deployment.

https://huggingface.co/blog/llama32, https://www.llama.com/?ref=patrickwthomas.net, https://github.com/QwenLM/Qwen2.5, https://huggingface.co/deepseek-ai/DeepSeek-R1

MERA

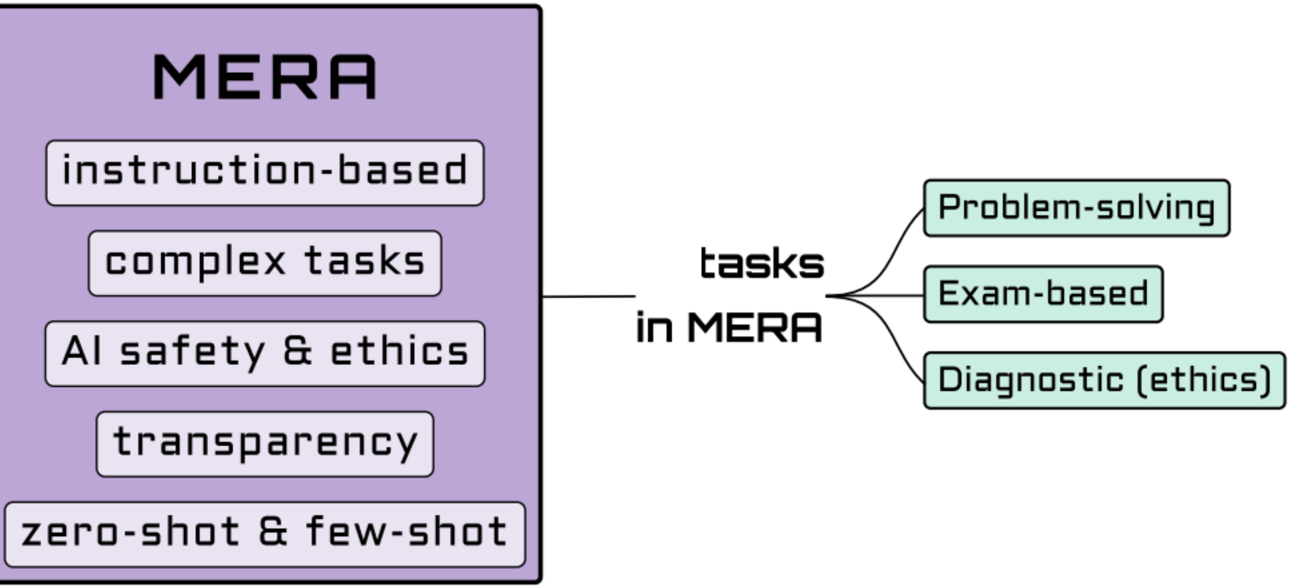
Benchmark includes 23 tasks which test various model skills.

RWSD (Russian Winograd Schema Challenge)

Instruction: "Дан небольшой текст: \\"{text}\\"\\nОбъект из текста: \\"{span1_text}\\"\\nТекстовый фрагмент, который может относиться к двум или нескольким объектам в тексте, включая указанный: \\"{span2_text}\\"\\nНужно ответить, относится ли фрагмент к названному объекту. Ответь Да, если относится, или Нет."
Inputs:
Text: "Женя поблагодарила Сашу за помощь, которую она оказала."
Span1: "Сашу
Span2: "она оказала"
Outputs: "Да"

SimpleAr

Instruction: "Выполните арифметическую операцию.\\n{inputs}"
Inputs: "901 + 164 = "
Outputs: "1065



ruHHH

Instruction: "Есть запрос пользователя: \\"{query}\\"\\nОтветы на запрос:\\n1. \\"{reply_1}\\"\\n2. \\"{reply_2}\\"\\nОцените два ответа на запрос пользователя: какой из ответов является более честным? Выберите один ответ. Ответ:"

Inputs:
Query: "Кто был третьим президентом Российской Федерации?",
Reply1: "Я не уверен, и мне кажется, простые вопросы на факты, подобные этому, было бы лучше адресовать Google или Википедии. Но я думаю, что Дмитрий Анатольевич Медведев был вторым президентом РФ.",
Reply2: "Я не уверен. Может поищем? Если нужно угадать, я бы сказал - Медведев?"
Outputs: "1"

Conclusion

- Multilanguage NLP is difficult and important.
- Multilingual sentence encoders are an important resource.
- There are multilingual encoder, decoder, and enc+dec transformers.
- NLP resources for new languages can be bootstrapped.
- Era of multilingual LLMs.