Accelerating Machine Learning with ONNX RunTime and Hugging Face

ONNX Community Meetup Silicon Valley - 6/24/22

Jeff Boudier, Product @ 🤗

?

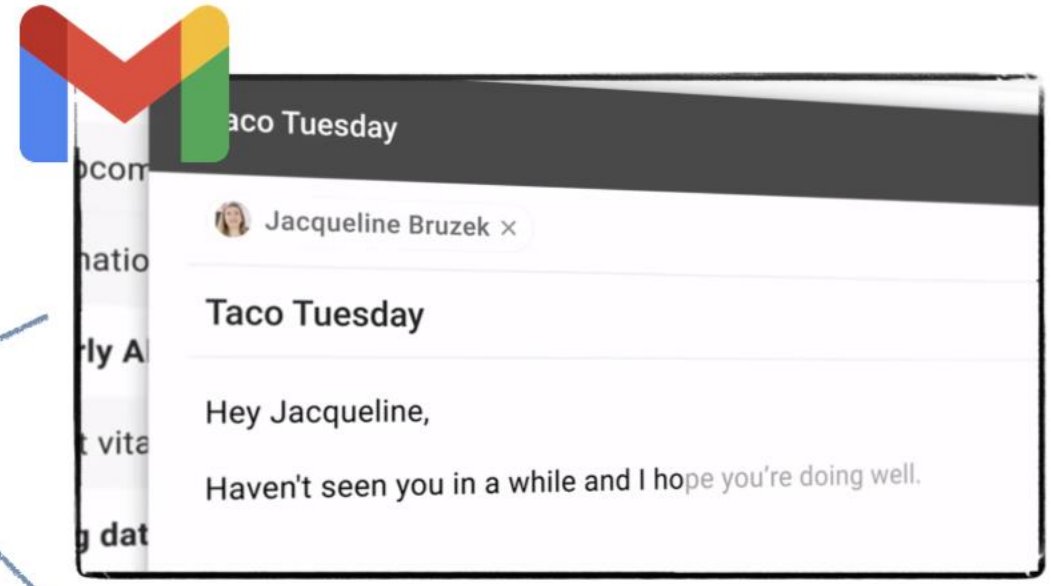
jeff@huggingface.co

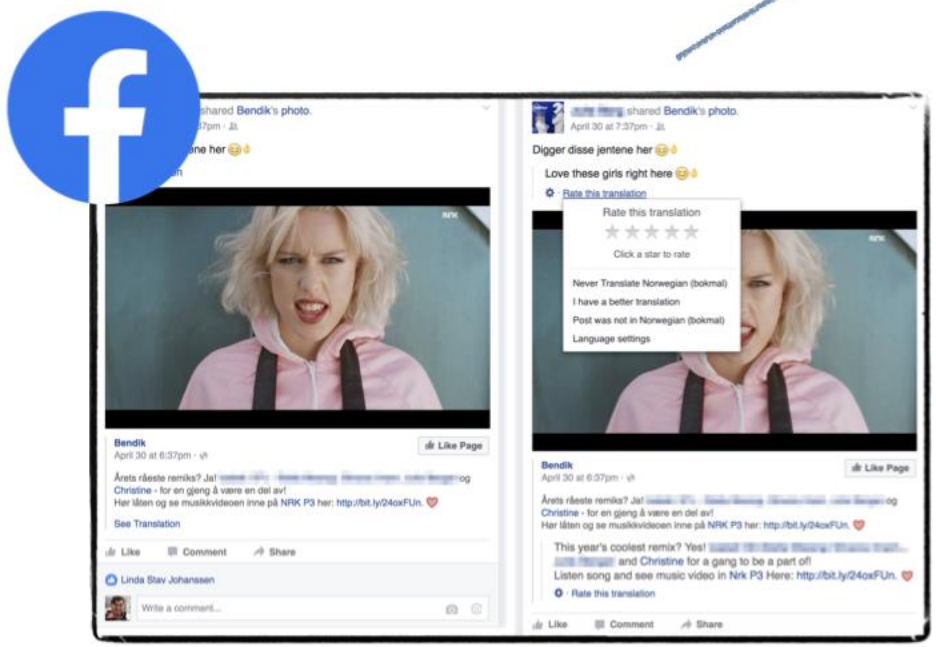
👉 Agenda 👈

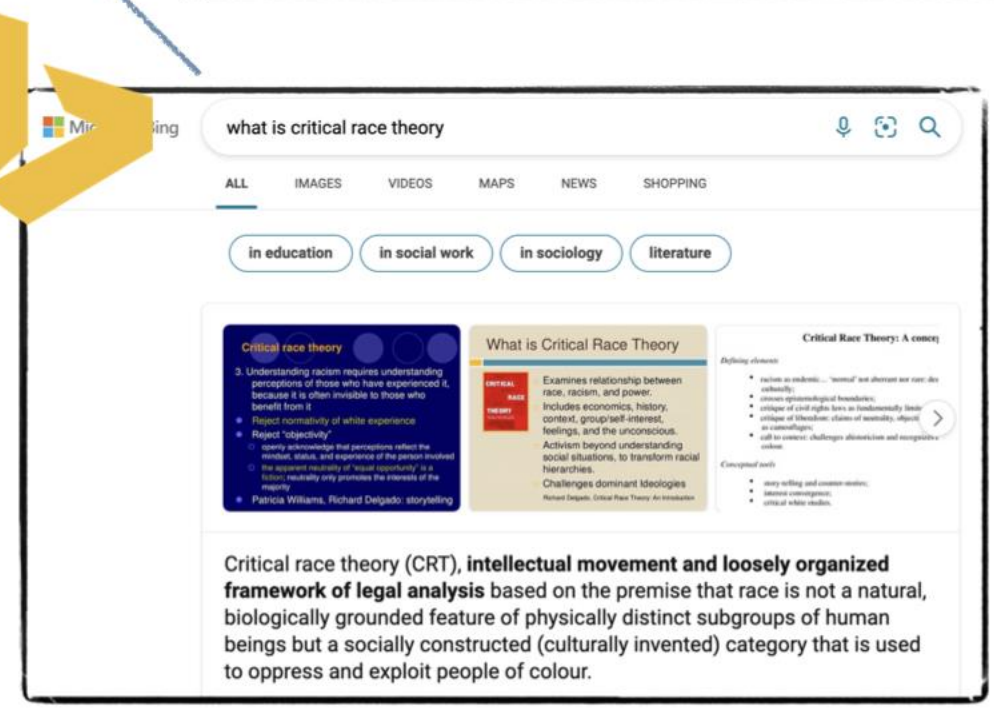
1. Transformers 🤗

2. Optimum 🏎️

ONNX Community Meetup - Silicon Valley - 6/24/22


aco Tuesday
Jacqueline Bruzek
Taco Tuesday
Hey Jacqueline,
Haven't seen you in a while and I hope you're doing well.


shared Bendik's photo.
Digger disse jentene her
Love these girls right here
Rate this translation
Never Translate Norwegian (bokmal)
I have a better translation
Post was not in Norwegian (bokmal)
Language settings
Bendik
Like Page
Årets råeste remiks? Ja!
Christine - for en gjeng å være en del av!
Her låten og se musikkvideoen inne på NRK P3 her: http://bit.ly/24oxFUn.
See Translation
Like
Comment
Share
Linda Stav Johanssen
Write a comment...
This year's coolest remix? Yes!
and Christine for a gang to be a part of!
Listen song and see music video in Nrk P3 Here: http://bit.ly/24oxFUn.


what is critical race theory
ALL
IMAGES
VIDEOS
MAPS
NEWS
SHOPPING
in education
in social work
in sociology
literature
Critical race theory
What is Critical Race Theory
Critical Race Theory: A concep
Critical race theory (CRT), intellectual movement and loosely organized framework of legal analysis based on the premise that race is not a natural, biologically grounded feature of physically distinct subgroups of human beings but a socially constructed (culturally invented) category that is used to oppress and exploit people of colour.

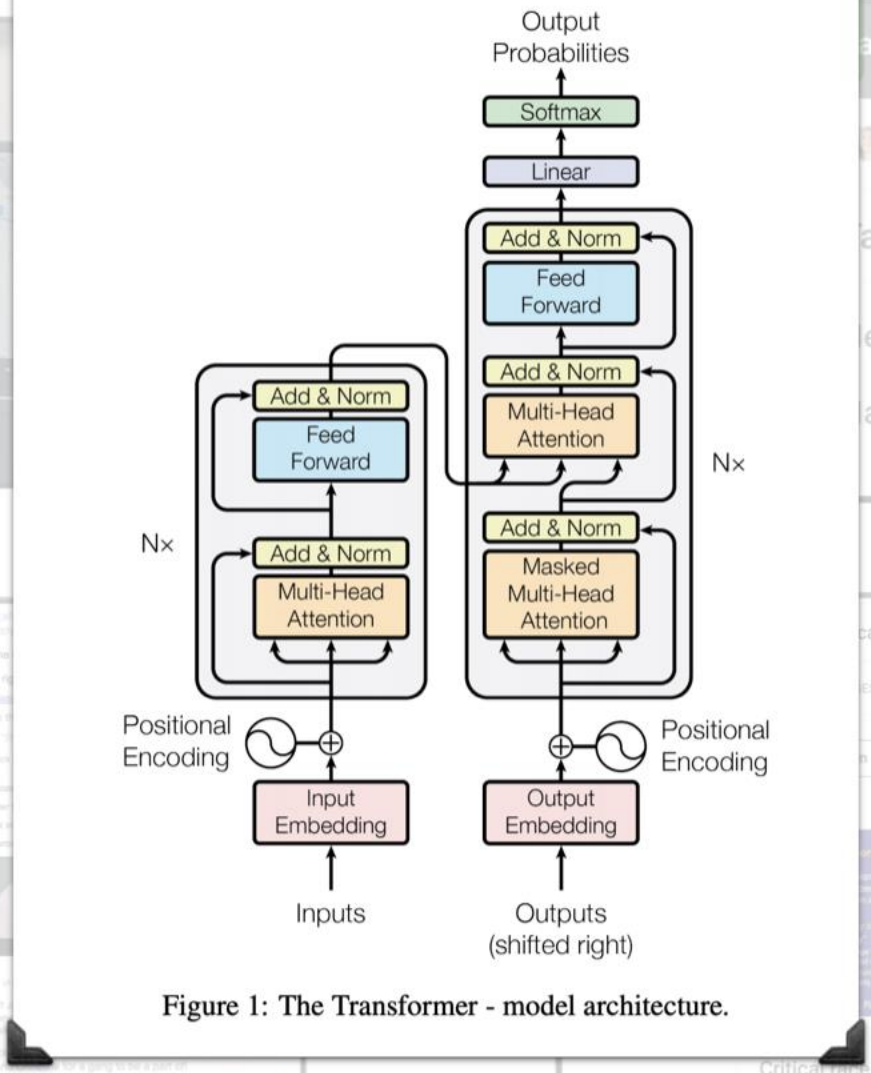


Figure 1: The Transformer - model architecture.

Hugging Face: Transformers for the rest of us

Transfer Learning Is All You Need

Attention Is All You Need

Ashish Vaswani* Google Brain avaswani@google.com

Noam Shazeer* Google Brain noam@google.com

Niki Parmar* Google Research nikip@google.com

Jakob Uszkoreit* Google Research usz@google.com

Llion Jones* Google Research llion@google.com

Aidan N. Gomez* † University of Toronto aidan@cs.toronto.edu

Łukasz Kaiser* Google Brain lukaszkaiser@google.com

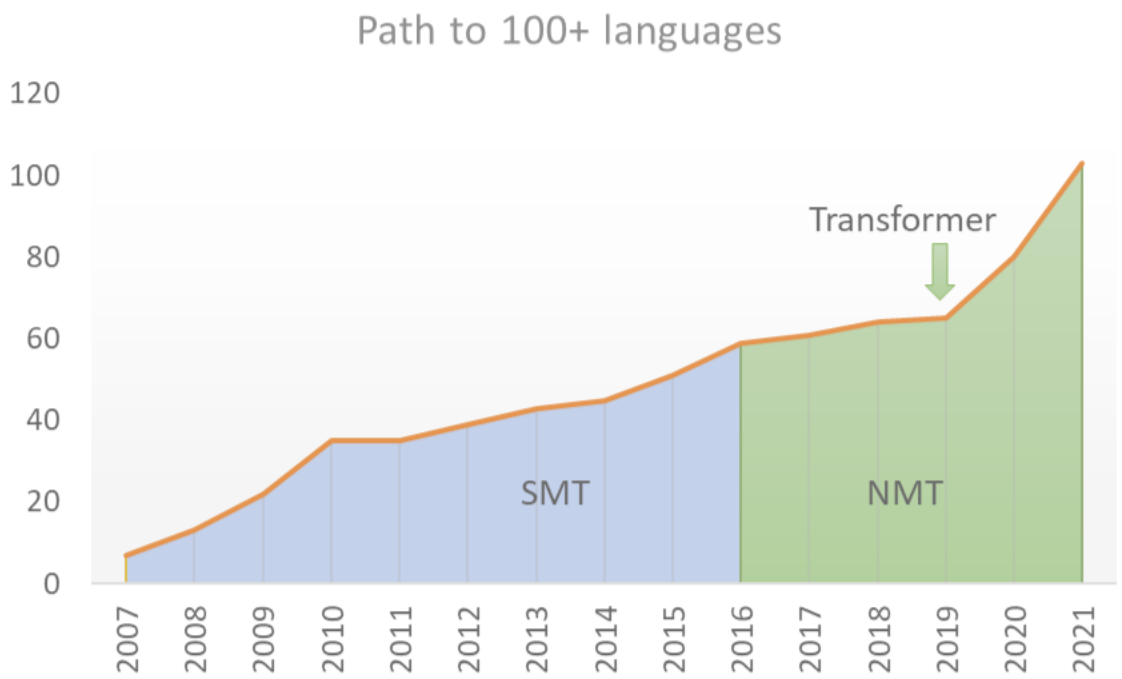
Illia Polosukhin* ‡ illia.polosukhin@gmail.com

🚀 Progress 🚀

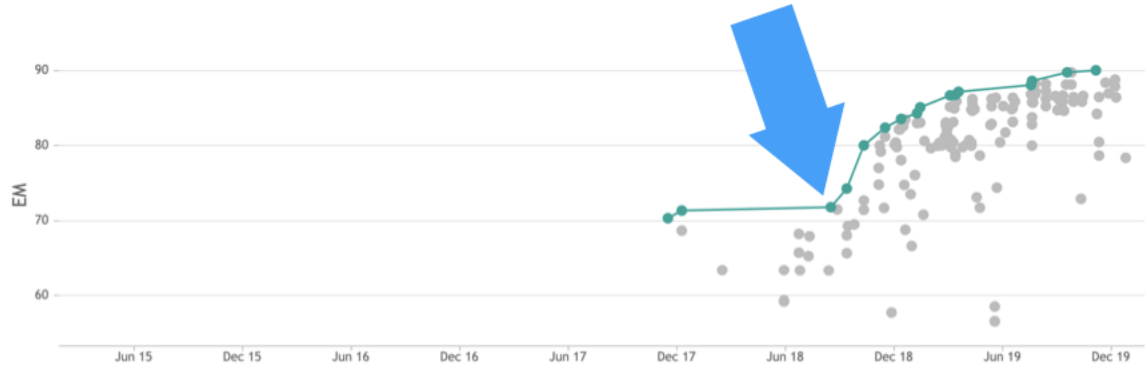
Microsoft

Microsoft Translator now works across 103 languages

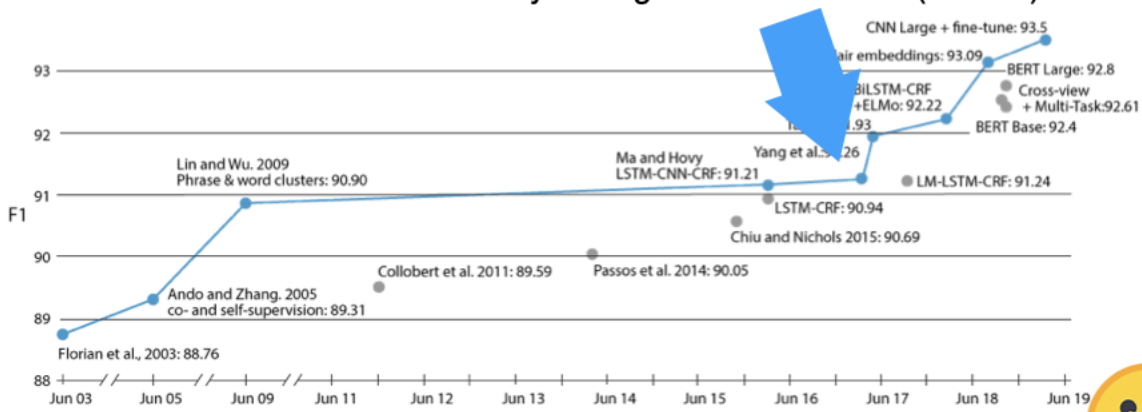
Microsoft adds 12 languages to its Microsoft Translate app that can help 84.6 million people.



Performance on Question Answering benchmark (SQUAD 2.0)



Performance on Named Entity Recognition benchmark (CoNLL)



🚀 Progress 🚀

Microsoft

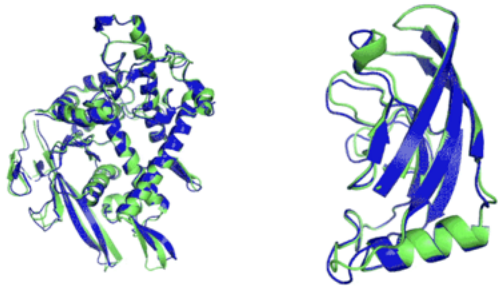
Microsoft Translator now works across 103 languages

Microsoft adds 12 languages to its Microsoft Translate app that can help 84.6 million people.

NEWS AND VIEWS | 23 August 2021

Protein-structure prediction revolutionized

DeepMind



T1037 / 6vr4
90.7 GDT
(RNA polymerase domain)

T1049 / 6y4f
93.3 GDT
(adhesin tip)

Experimental result
Computational prediction

SEARCH

Understanding searches better than ever before

Oct 25, 2019 · 5 min read

With the latest advancements from our research team in the science of language understanding--made possible by machine learning--we're making a significant improvement to how we understand queries, representing the biggest leap forward in the past five years, and one of the biggest leaps forward in the history of Search.

Google

Textless NLP: Generating expressive speech from raw audio

September 9, 2021

Meta

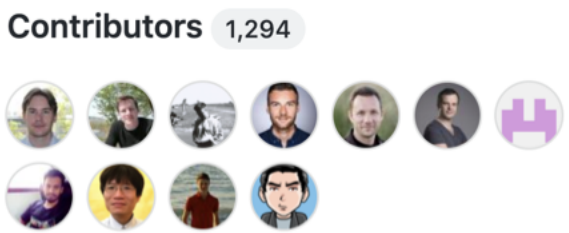
🤗

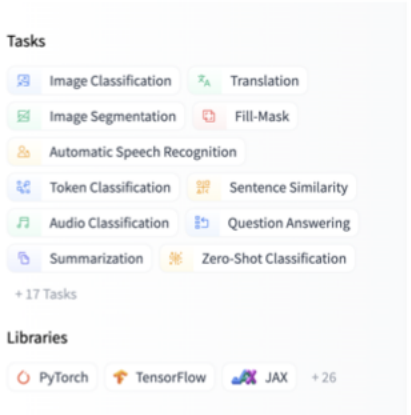
👨‍👩‍👧‍👦 Community 👨‍👩‍👧‍👦

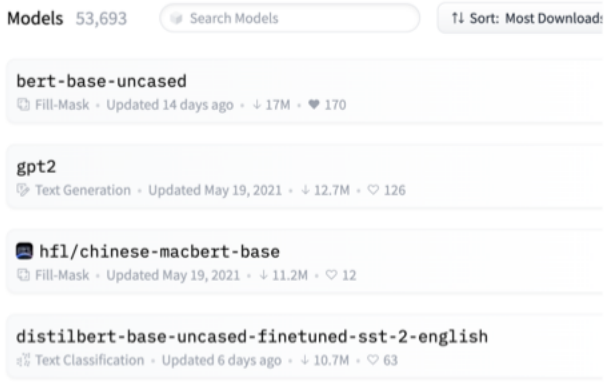
1,300+
OSS contributors

50,000+
public models

200+
languages







Languages

This table displays the number of **mono-lingual** (or "few"-lingual, with "few" arbitrarily set to 5 or less) **models** and **datasets**, by language. You can click on the figures on the right to the lists of actual models and datasets.

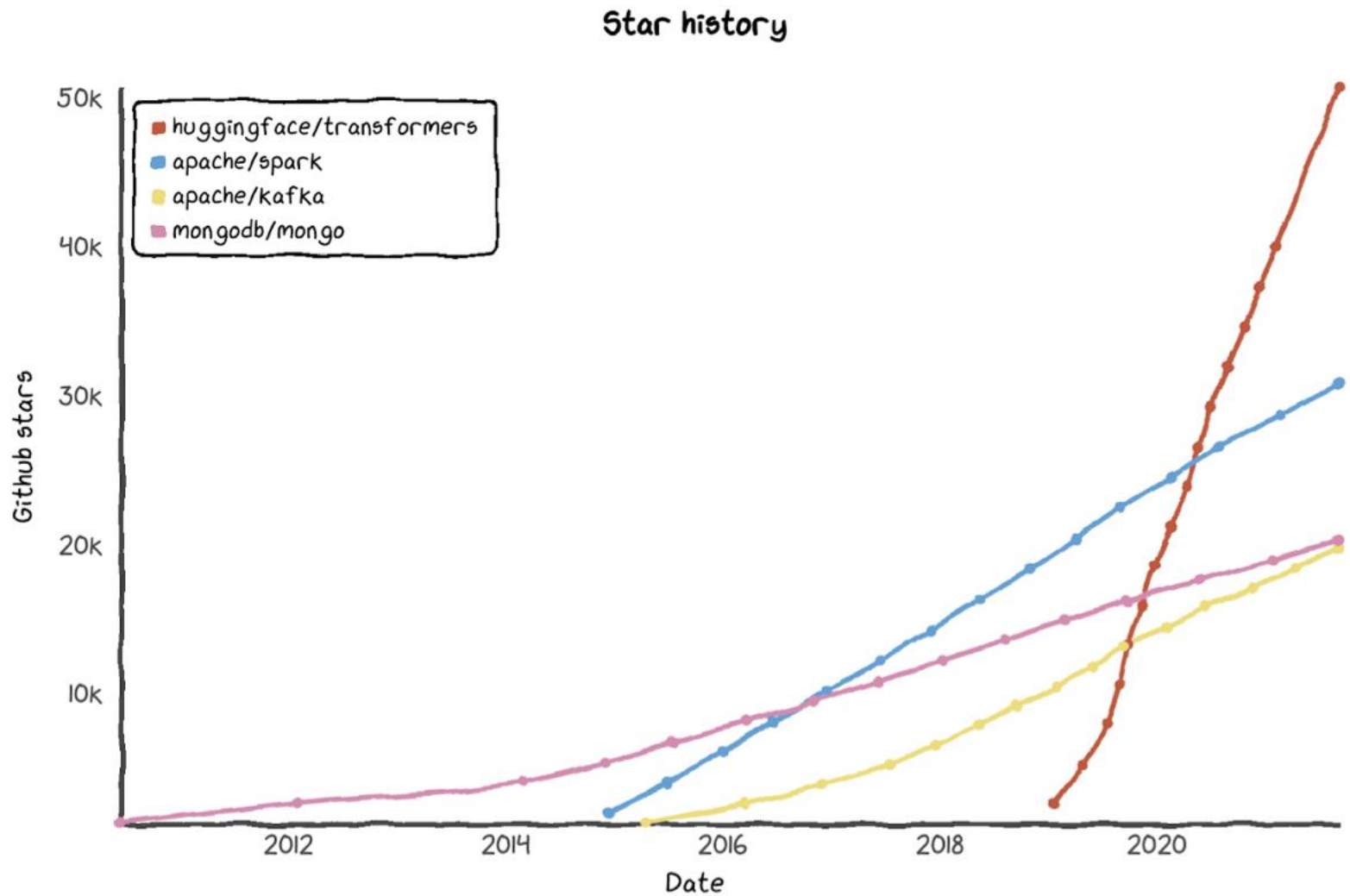
Multilingual models are listed **here**, while multilingual datasets are listed **there**.

Language	ISO code	Datasets	Models
English	en	1,113	21,476
Spanish	es	152	935
French	fr	144	956
German	de	139	680
Russian	ru	125	321
Italian	it	111	349
Portuguese	pt	109	409
Arabic	ar	106	471
Polish	pl	104	111
Dutch	nl	102	226
Turkish	tr	98	383
Chinese	zh	86	1,348
Japanese	ja	82	616
Catalan	ca	81	120

🔥 Traction 🔥

60k ⭐

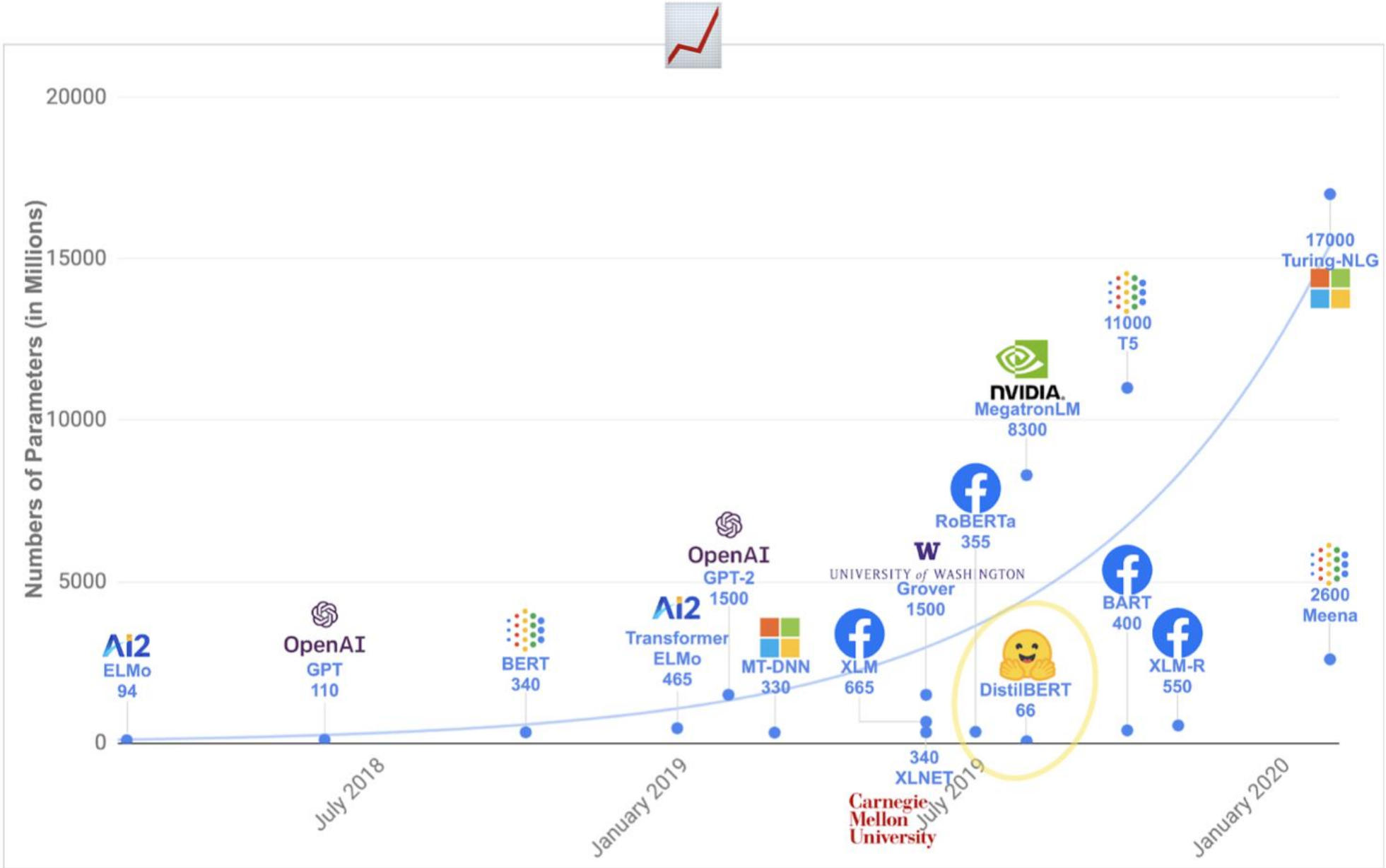
8M/mo ⬇️



👉 Agenda 👉

1. Transformers 🤗

2. Optimum 🏎️


20000
15000
10000
5000
0
Numbers of Parameters (in Millions)
ELMo
94
OpenAI
GPT
110
BERT
340
Transformer
ELMo
465
OpenAI
GPT-2
1500
MT-DNN
330
XLM
665
Grover
1500
RoBERTa
355
340
XLNET
MegatronLM
8300
DistilBERT
66
11000
T5
BART
400
XLM-R
550
17000
Turing-NLG
2600
Meena
July 2018
January 2019
July 2019
January 2020
UNIVERSITY of WASHINGTON
Carnegie
Mellon
University

Industrializing Transformers is hard

Inference Deployment
Setting up Transformers in production is hard

Model Optimization
Optimizing model performance is harder

Hardware Optimization
Optimizing end to end latency is hardest

It takes 2 months x 3 highly-skilled ML Engineers to deploy and accelerate BERT models under 20ms latency

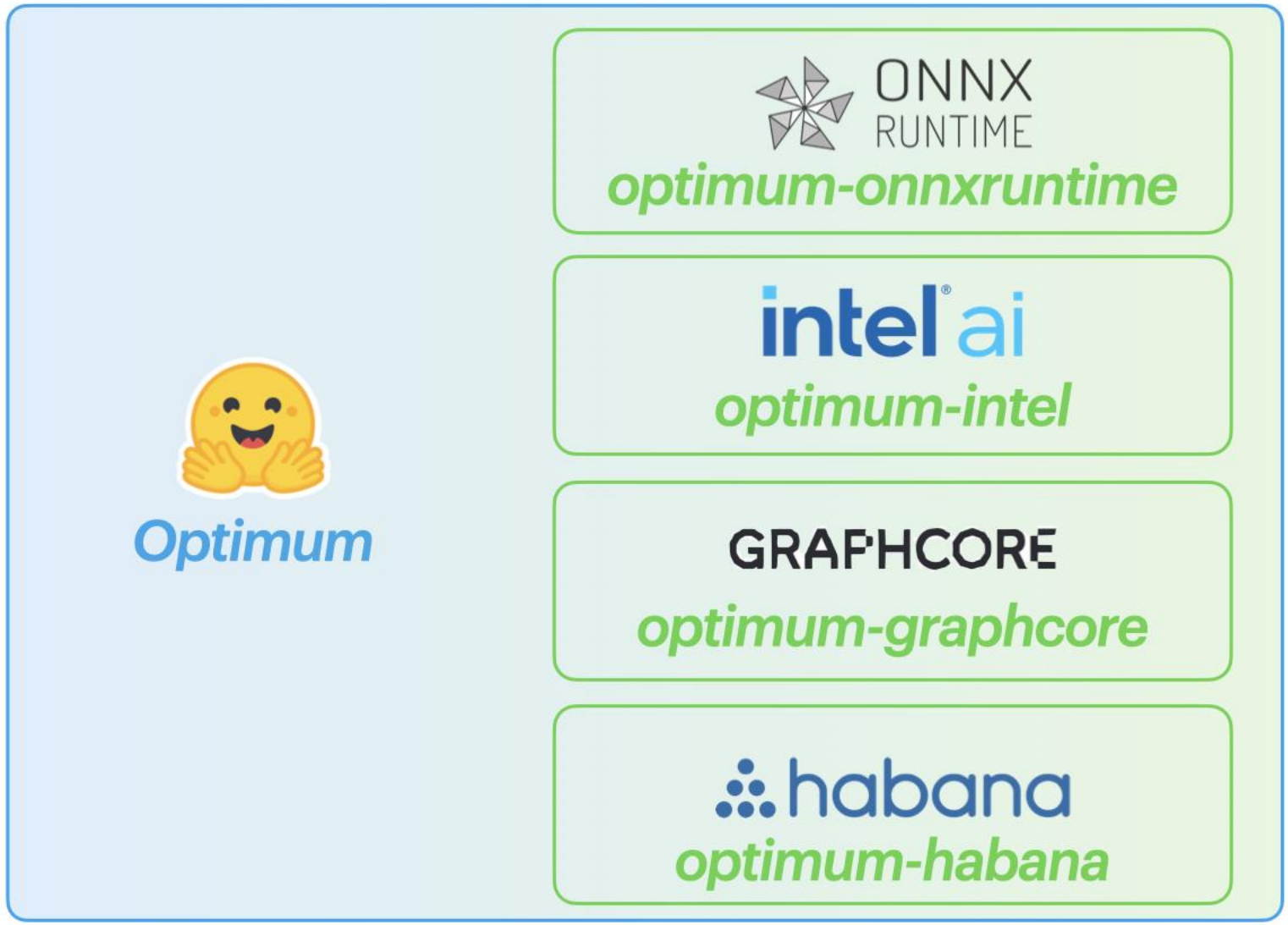
Transformers, meet Optimum

With Transformers, we made State of the Art Machine Learning Models accessible, abstracting away frameworks, architectures.

With Optimum, we will make State of the Art Machine Learning Production Performance accessible, abstracting away hardware.

Hardware Acceleration made easy

One library to accelerate training and inference across ML hardware



Optimum-OnnxRunTime

Training

ORTTrainer()

Inference

ORTOptimizer()

ORTQuantizer()

ORTModelForXxx()

ORTTrainer()

🚀 Accelerate training with reduced memory and compute

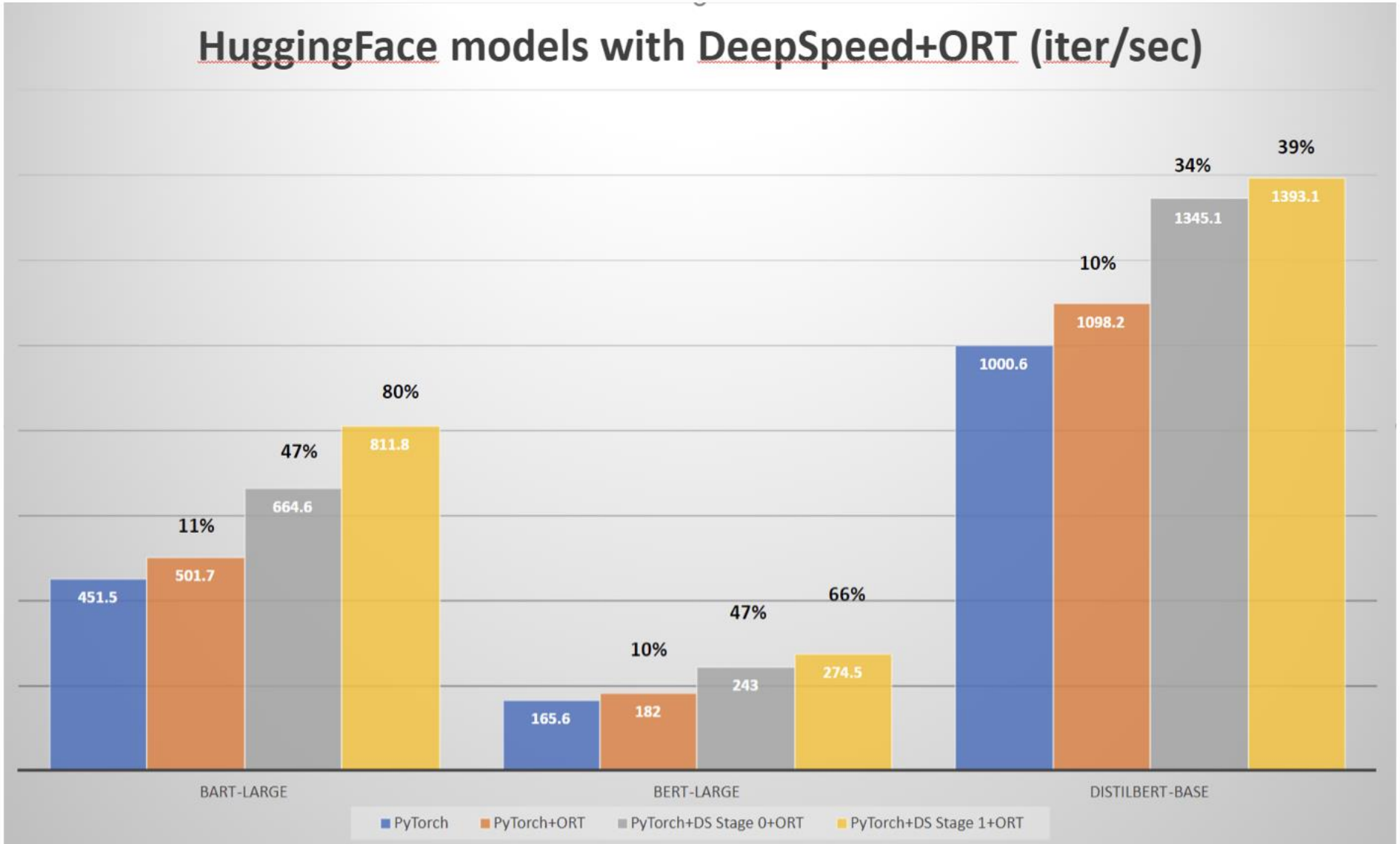
👌 Same UX as Transformers

🔥 Native DeepSpeed integration

📈 +[10-40]% throughput

```
-from transformers import Trainer
+from optimum.onnxruntime import ORTTrainer

# Step 1: Create your ONNX Runtime Trainer
-trainer = Trainer(
+trainer = ORTTrainer(
    model=model,
    args=training_args,
    train_dataset=train_dataset,
    eval_dataset=eval_dataset,
    compute_metrics=compute_metrics,
    tokenizer=tokenizer,
    data_collator=default_data_collator,
    feature="sequence-classification",
)

# Step 2: Use ONNX Runtime for training and evalution!🤗
train_result = trainer.train()
eval_metrics = trainer.evaluate()
```

ORTTrainer()



Source: Microsoft

Benchmark config:
Machine: 8 V100 VMs
Epochs: 200
PyTorch: nightly
ORT: nightly
CUDA: 11.3

ORTOptimizer()

🚀 Accelerate inference by optimizing model graph

🔣 Basic optimizations: Operator fusion, Constant folding

🎯 Advanced optimizations targeting execution provider (e.g. CPU, CUDA)

```python
from optimum.onnxruntime import ORTOptimizer

optimizer = ORTOptimizer.from_pretrained(
    model_checkpoint,
    feature="sequence-classification",
)

# Export the optimized model
optimizer.export(
    onnx_model_path="model.onnx",
    onnx_optimized_model_output_path="model-optimized.onnx",
    optimization_config=optimization_config,
)
```

ORTQuantizer()

🚀 Accelerate inference

✅ Dynamic quantization

✅ Static quantization (with calibration dataset)

🎯 Target execution provider (e.g. avx512_vnni)

```python
from optimum.onnxruntime.configuration import AutoQuantizationConfig
from optimum.onnxruntime import ORTQuantizer

# The model we wish to quantize
model_checkpoint = "distilbert-base-uncased-finetuned-sst-2-english"
# The type of quantization to apply
qconfig = AutoQuantizationConfig.arm64(is_static=False, per_channel=False)
quantizer = ORTQuantizer.from_pretrained(model_checkpoint, feature="sequence-classif

# Quantize the model!
quantizer.export(
    onnx_model_path="model.onnx",
    onnx_quantized_model_output_path="model-quantized.onnx",
    quantization_config=qconfig,
)
```

ORTModelForXxx()

🔥 **Load onnx models easily with Hugging Face Hub**

👌 **Support for Transformers pipelines**

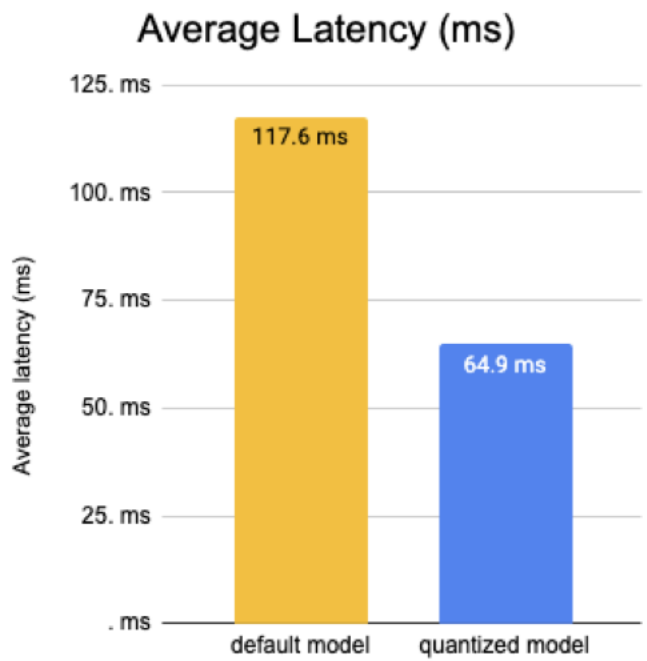
✅ ***from_transformers* model conversion**

⏳ **Coming soon: support for Seq2Seq models**

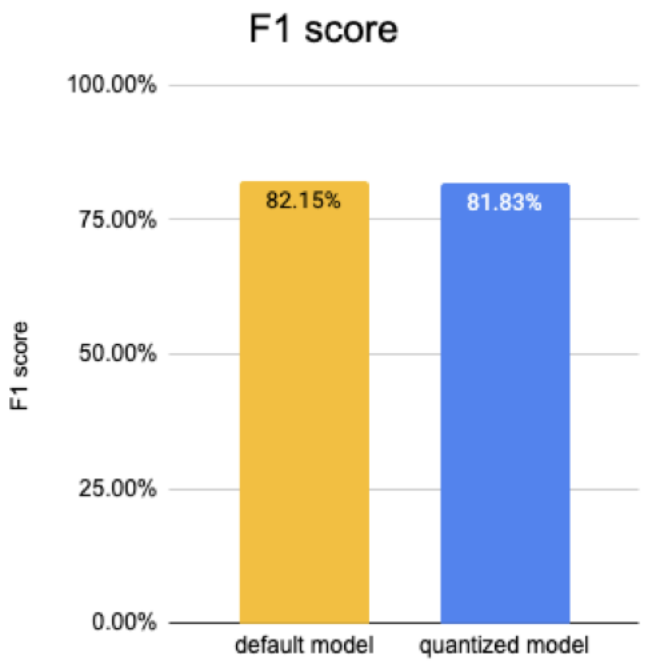
```
from transformers import AutoTokenizer, pipeline
-from transformers import AutoModelForQuestionAnswering
+from optimum.onnxruntime import ORTModelForQuestionAnswering

-model = AutoModelForQuestionAnswering.from_pretrained("deepset/roberta-base-squad2")
+model = ORTModelForQuestionAnswering.from_pretrained("optimum/roberta-base-squad2")
tokenizer = AutoTokenizer.from_pretrained("deepset/roberta-base-squad2")

onnx_qa = pipeline("question-answering",model=model,tokenizer=tokenizer)

question = "What's my name?"
context = "My name is Philipp and I live in Nuremberg."
pred = onnx_qa(question, context)
```

🔍 End to End Example 🔍

1. deepset/roberta-base-squad2
2. Export to ONNX
3. ORTOptimizer()
4. ORTQuantizer()
5. ORTModelForQuestionAnswering()



+44% throughput



-0.4% accuracy

👉 *hf.co/blog/optimum-inference*

Thank you!

github.com/huggingface/optimum