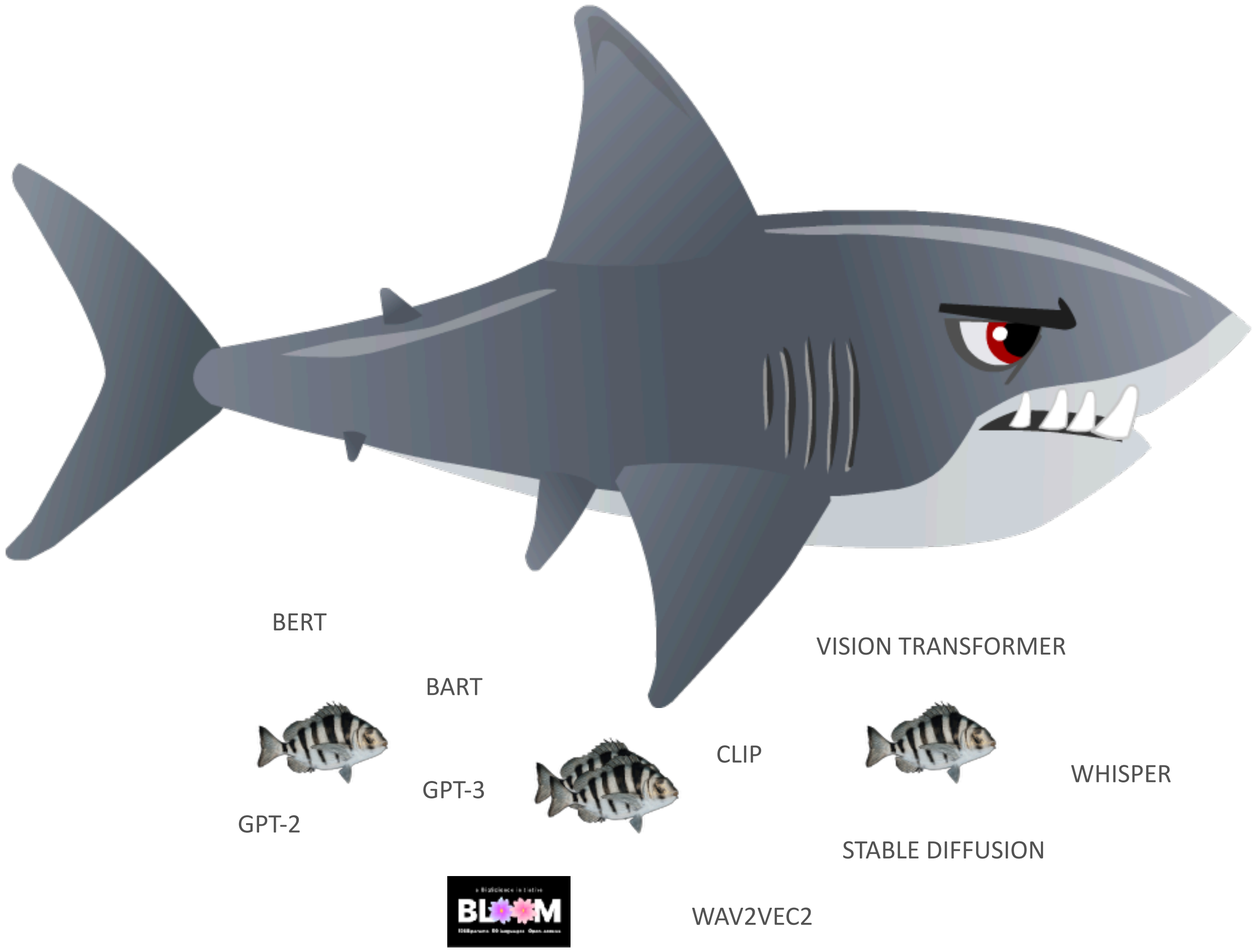
Hugging Face and Intel

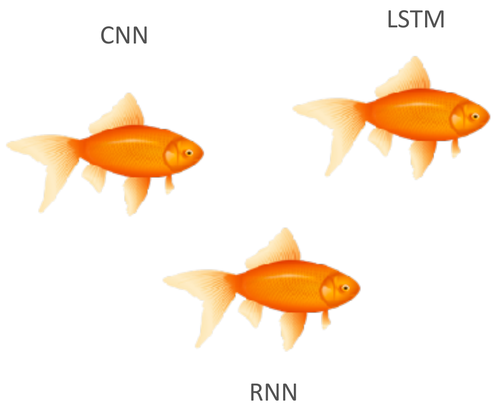
partnering to democratize ML hardware acceleration

Julien Simon, Chief Evangelist, Hugging Face

intel.

2022: Transformers are eating Deep Learning





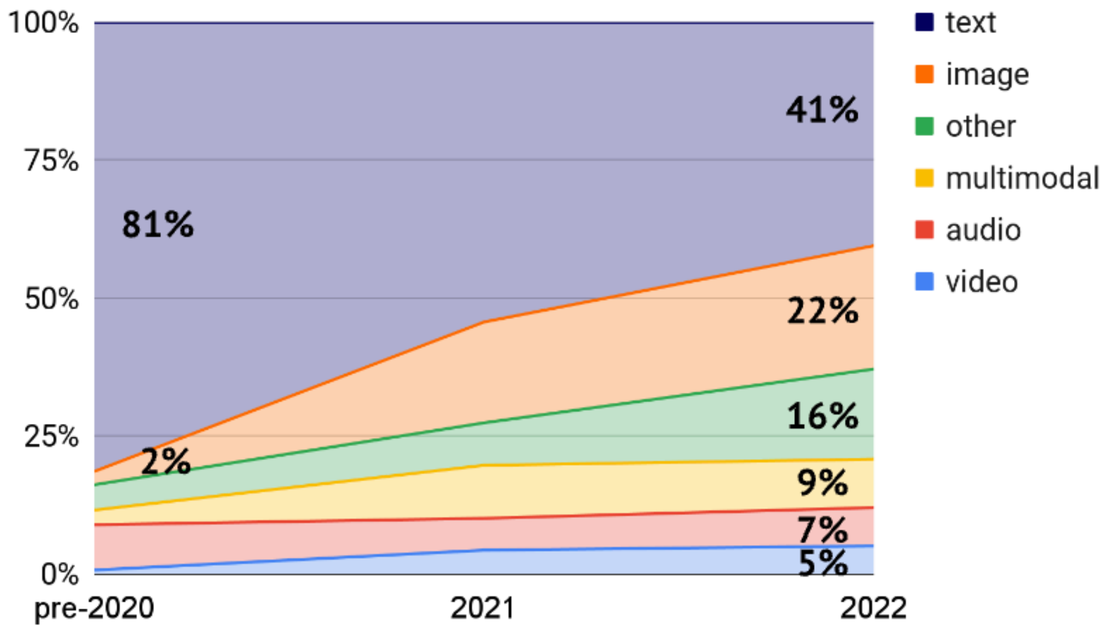
"*Transformers are emerging as a general-purpose architecture for ML*"
https://www.stateof.ai (2021)

RNN and CNN usage down, Transformers usage up!
https://www.kaggle.com/kaggle-survey-2021

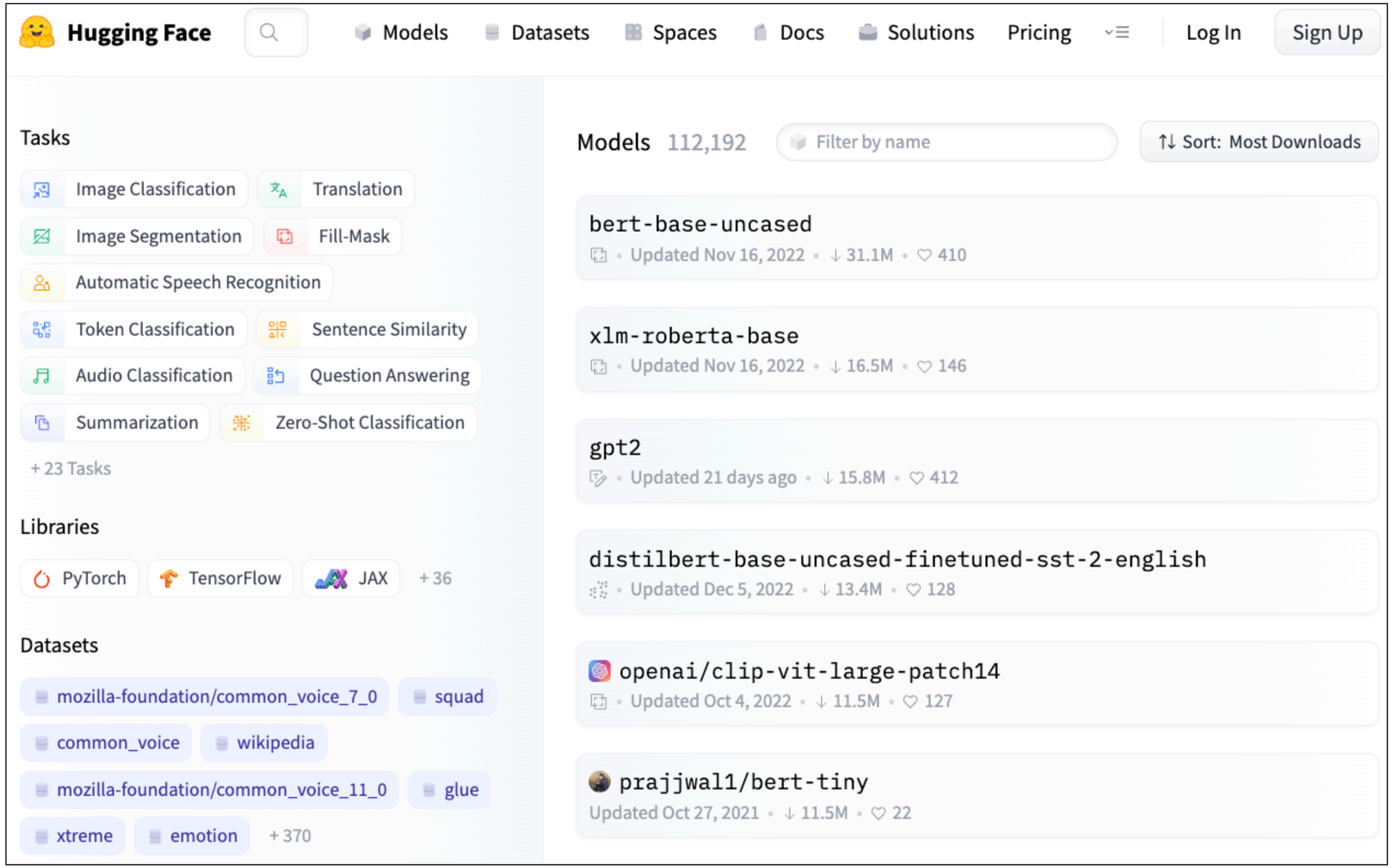
All modalities, and multi-modal too

Transformers are becoming truly cross-modality

- **In the 2020 State of AI Report we predicted that transformers would expand beyond NLP to achieve state of the art in computer vision. It is now clear that transformers are a candidate general purpose architecture. Analysing transformer-related papers in 2022 shows just how ubiquitous this model architecture has become.**



The Hugging Face Hub: The Github of Machine Learning



https://huggingface.co

150K models

24K datasets

25+ ML libraries: Keras, spaCY, Scikit-Learn, fastai, etc.

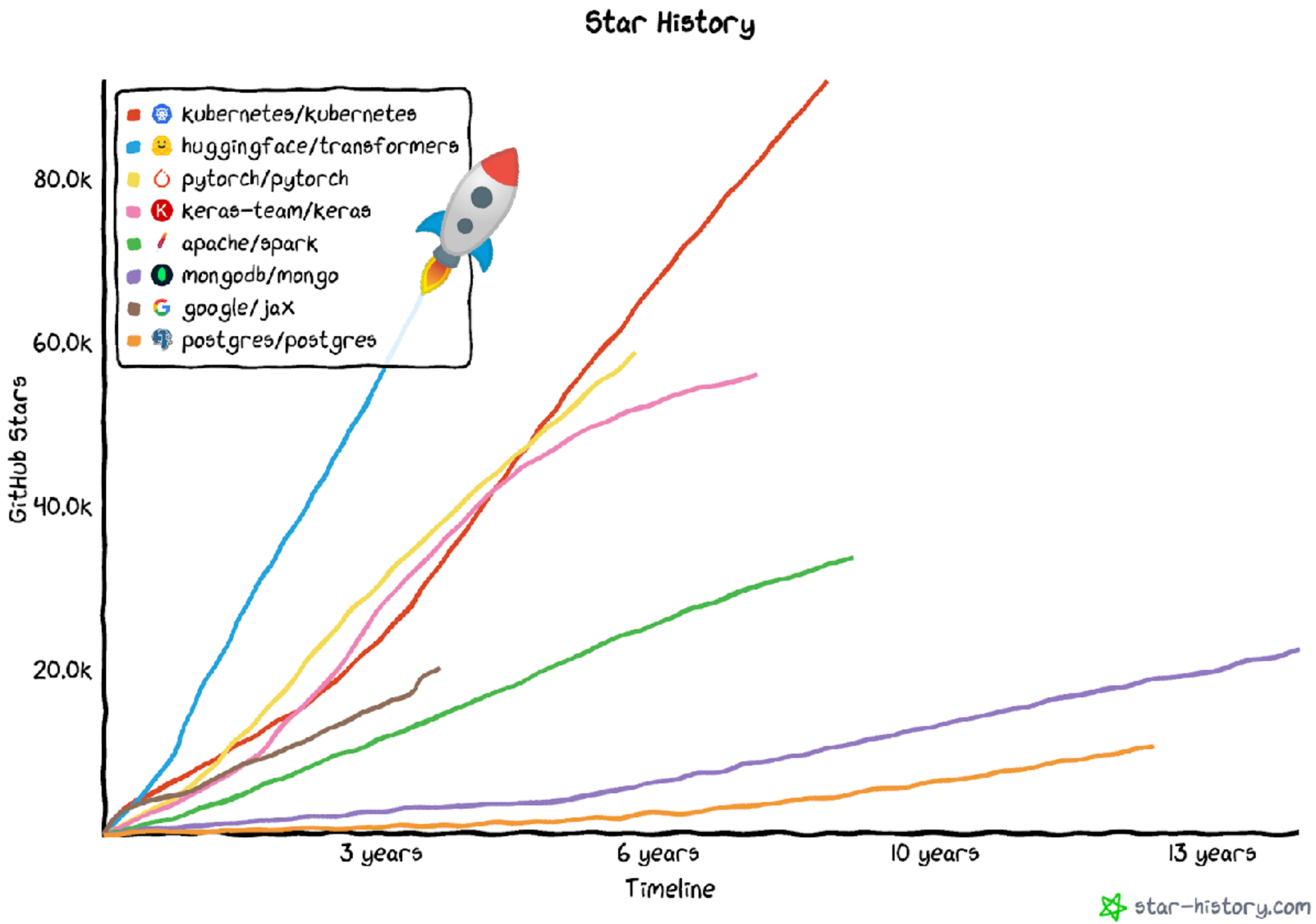
10K organizations

100K+ users daily

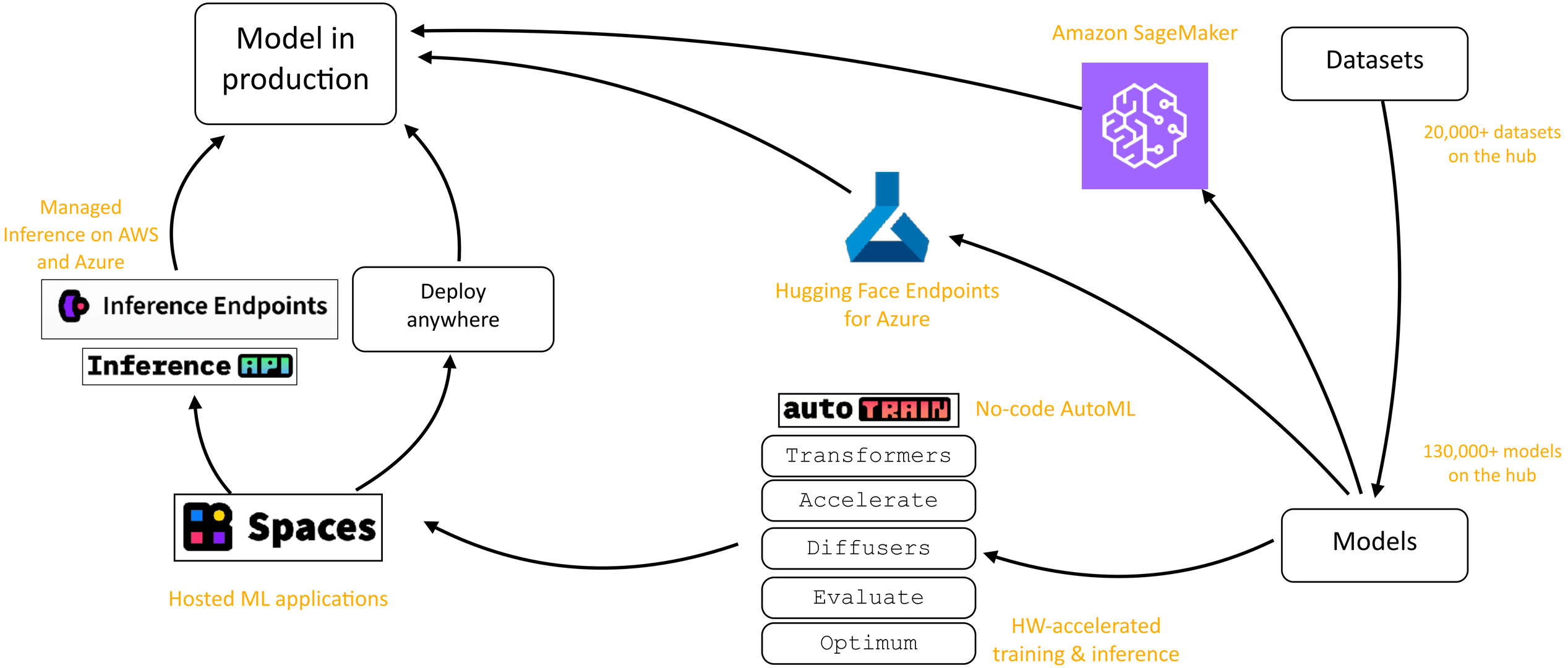
1M+ downloads daily

Hugging Face: one of the fastest-growing open source projects

https://github.com/huggingface/transformers/



Hugging Face at a glance



Partnering to Democratize ML Hardware Acceleration

× intel ai

https://huggingface.co/blog/intel

Hugging Face and Intel partnership

- Accelerate Transformer fine-tuning and inference on Intel platforms
- Deliver best-in-class cost-performance and scalability
- Build the simplest developer experience
- Share results with the Open Source community

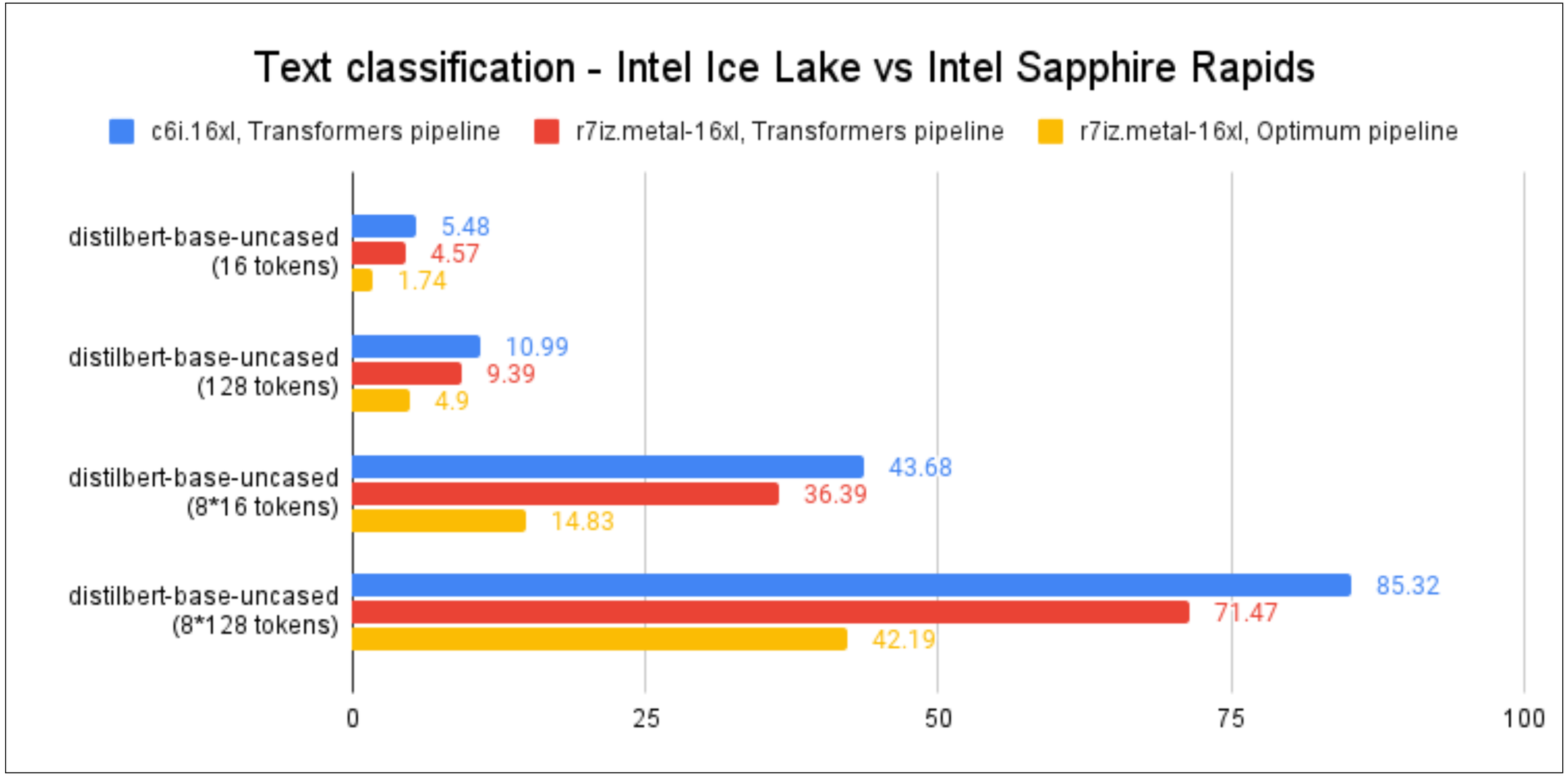
Accelerate Hugging Face transformers with Optimum Intel

https://github.com/huggingface/optimum-intel

- Open source library, bringing Intel acceleration to the Hugging Face Transformers library
  - Hardware: AVX, AMX
  - Software: Intel Extension for PyTorch (IPEX), Intel Neural Compressor, Intel OpenVINO
- Acceleration, quantization and pruning in just a few lines of Python!

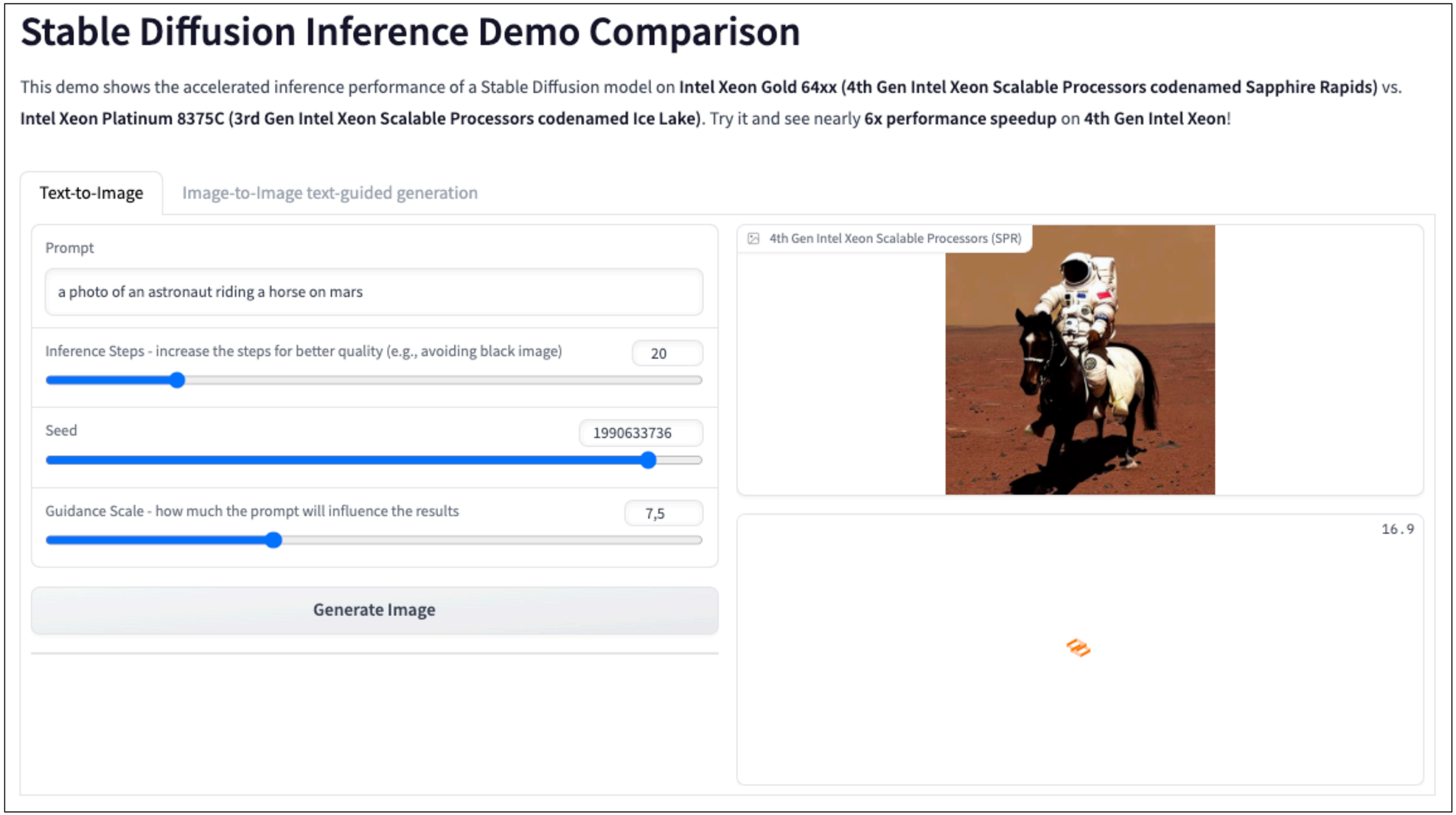
```
from transformers import AutoModelForQuestionAnswering
from neural_compressor.config import PostTrainingQuantConfig
from optimum.intel.neural_compressor import INCQuantizer

model_name = "distilbert-base-cased-distilled-squad"
model = AutoModelForQuestionAnswering.from_pretrained(model_name)
# The directory where the quantized model will be saved
save_dir = "quantized_model"
# Load the quantization configuration detailing the quantization we wish to apply
quantization_config = PostTrainingQuantConfig(approach="dynamic")
quantizer = INCQuantizer.from_pretrained(model)
# Apply dynamic quantization and save the resulting model
quantizer.quantize(quantization_config=quantization_config, save_directory=save_dir)
```

Inference on Intel Sapphire Rapids with Optimum Intel and IPEX



https://huggingface.co/blog/intel-sapphire-rapids-inference

Text-to-image generation with Stable Diffusion on Sapphire Rapids



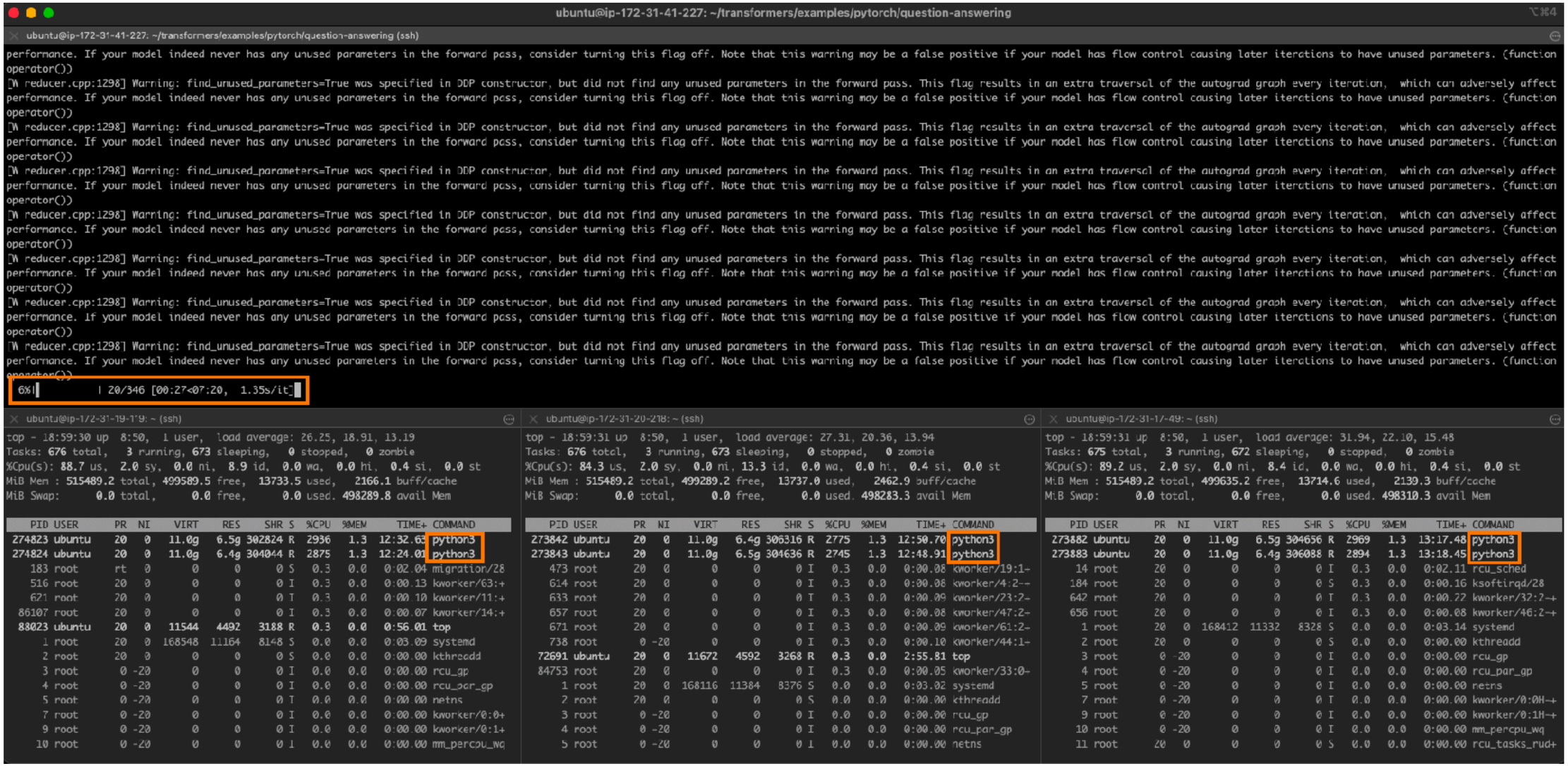
https://huggingface.co/spaces/Intel/Stable-Diffusion-Side-by-Side

Few-shot learning on Intel Sapphire Rapids with IPEX

```
Dataset({
    features: ['text', 'label'],
    num_rows: 16
})
Dataset({
    features: ['text', 'label'],
    num_rows: 38000
})
model_head.pkl not found on HuggingFace Hub, initialising classification head wi
th random weights. You should TRAIN this model on a downstream task to use it fo
r predictions and inference.
***** Running training *****
  Num examples = 512
  Num epochs = 1
  Total optimization steps = 32
  Total train batch size = 16
Iteration: 100%|                                        | 32/32 [02:23<00:00,  4.49s/it]
Epoch: 100%|                                           | 1/1 [02:23<00:00, 143.77s/it]
***** Running evaluation *****
{'accuracy': 0.9459736842105263}
```

https://huggingface.co/blog/setfit

Distributed training on Intel Sapphire Rapids with IPEX and oneCCL



https://huggingface.co/blog/intel-sapphire-rapids

Getting started

Stay in touch!

@julsimon
julsimon.medium.com
youtube.com/c/juliensimonfr

https://huggingface.co/intel

https://huggingface.co/hardware/intel

https://huggingface.co/docs/optimum/intel/index

https://github.com/huggingface/optimum-intel

https://www.intel.com/content/www/us/en/developer/partner/hugging-face.html

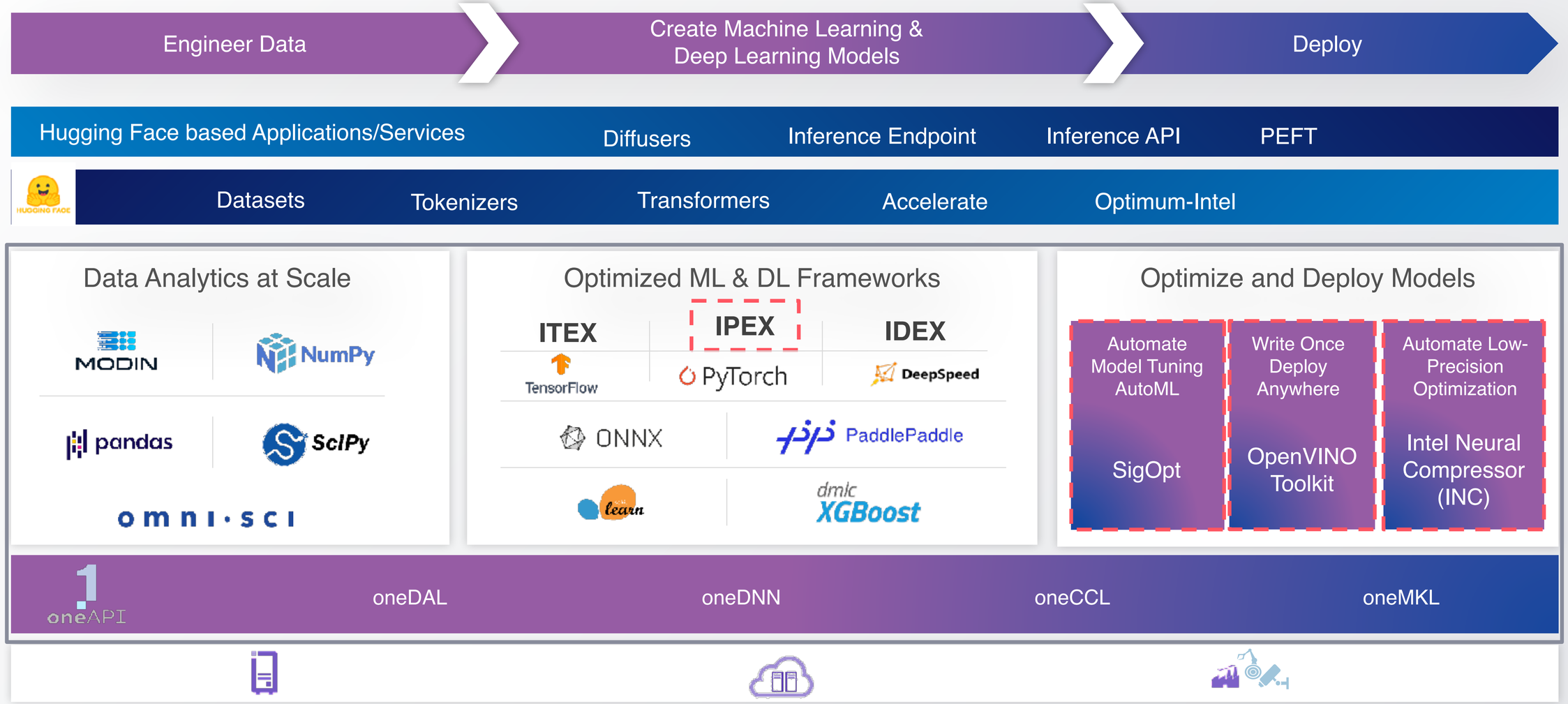
Hugging Face and Intel

partnering to democratize ML hardware acceleration

Matrix Yao, Lead AI Software Architect, Intel Corporation

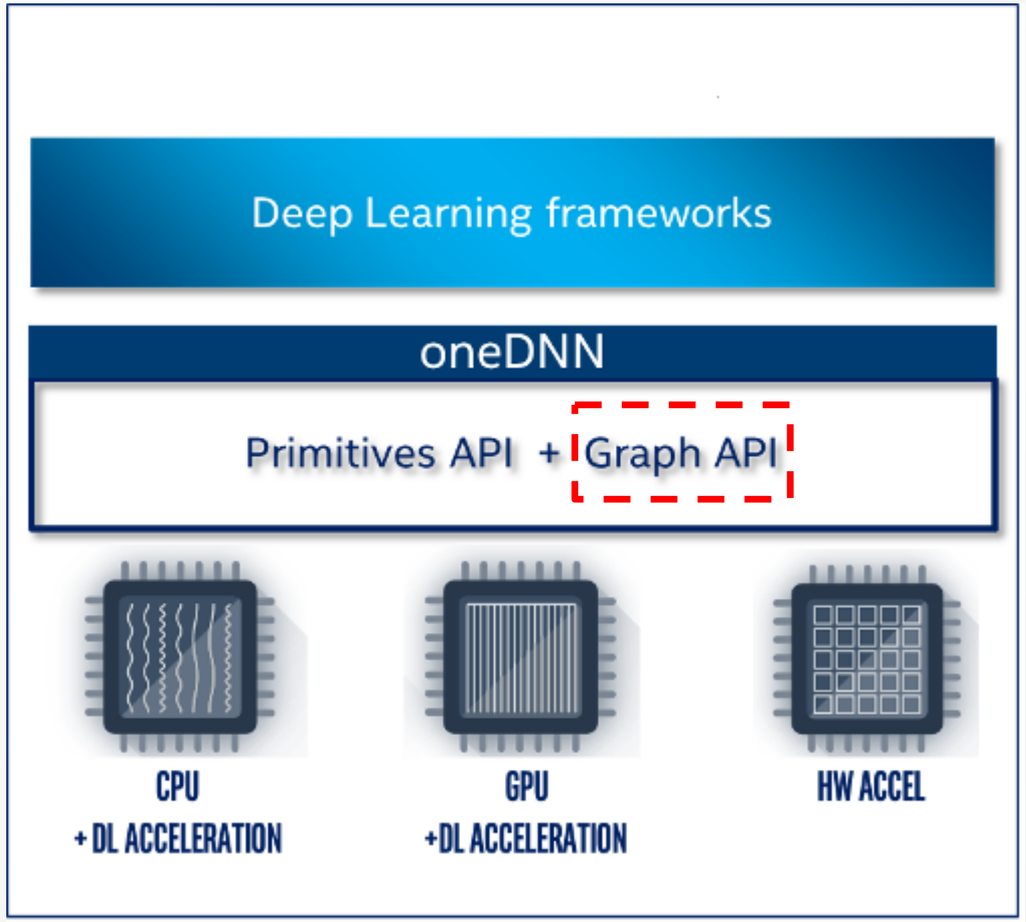
intel®

Intel Software Ecosystem deeply integrated w/ Hugging Face

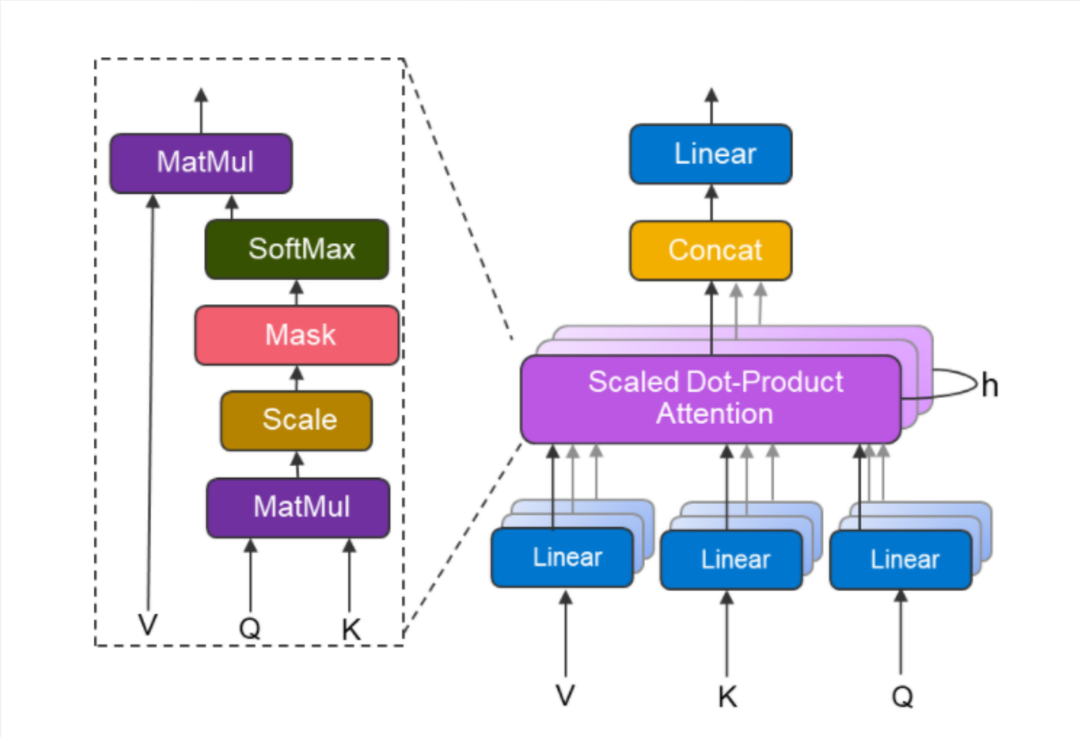


Build *OOB is the best performance* Hugging Face experience on Intel platforms

Enjoy Intel SW Optimization Benefit with Zero Code Change



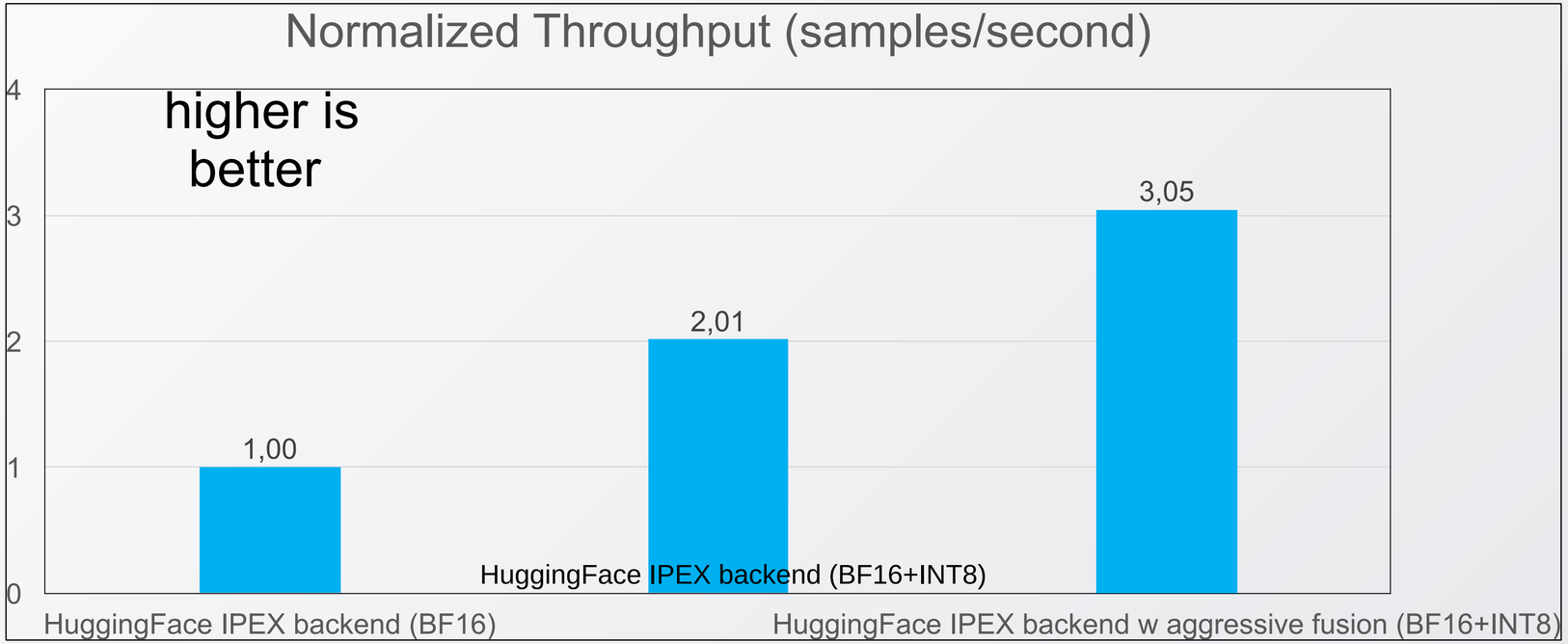
oneDNN-Graph brings INT8 MHA aggressive fusion capability and is integrated to IPEX.



reshape + permute + transpose + bmm + mul + add + softmax + quantize + (bmm + transpose + reshape + quantize)

fuse to single op in runtime

Enjoy Intel SW Optimization Benefit with Zero Code Change



* BERT-Large-uncased; SQUAD-v1.1; seq len=384, batch size per core = 1, single socket
* Intel(R) Xeon(R) Platinum 8480+ 56 cores on Dennard Pass platform and software with 512GB memory (16x32GB DDR5 4800 MT/s [4800 MT/s]), microcode 0x90000c0, HT on, Turbo on, Rocky Linux 8.7, 4.18.0-372.32.1.el8_6.crt2.x86_64, 931.5G SSD. Multiple nodes connected with 200Gbps Mellanox EDR. PyTorch 1.13.1, IPEX 1.13.100, Transformers 4.26.1, oneDNN 2.6.0

Just pip install the new IPEX package, **zero code change**, HF users can enjoy oneDNN-Graph performance boost.