Product ⌄ Solutions ⌄ Open Source ⌄ Pricing

Search or jump to... /

Sign in Sign up

huggingface / transformers Public

Notifications Fork 23.6k Star 118k

Code Issues 701 Pull requests 222 Actions Projects 25 Security Insights

main ⌄ 303 branches 144 tags Go to file Code ⌄

stas00 small typo (#28229) ✕ 3cefac1 5 days ago 14,816 commits

| | | |
|---|---|---|
| .circleci | Fix the check of models supporting FA/SDPA not run (#28202) | last week |
| .github | Fix AMD push CI not triggered (#28029) | 2 weeks ago |
| docker | [`Awq`] Enable the possibility to skip quantization for some target m... | last week |
| docs | Fixing visualization code for object detection to support both types ... | last week |
| examples | Dev version | 3 weeks ago |
| model_cards | Update URL for Hub PR docs (#17532) | last year |
| notebooks | Update README.md (#25941) | 3 months ago |
| scripts | [`Styling`] stylify using ruff (#27144) | last month |
| src/transformers | small typo (#28229) | 5 days ago |
| templates | Fix [`input_embeds`] docstring in encoder-decoder architectures (#28... | last week |
| tests | [`Awq`] Enable the possibility to skip quantization for some target m... | last week |
| utils | Fix the check of models supporting FA/SDPA not run (#28202) | last week |
| .coveragerc | GPU text generation: mMoved the encoded_prompt to correct device | 4 years ago |
| .gitattributes | Add trajectory transformer (#17141) | last year |
| .gitignore | 🚨🚨 🚨🚨 [`Tokenizer`] attemp to fix add_token issues🚨🚨 🚨🚨 (#2... | 3 months ago |
| CITATION.cff | Update CITATION.cff (#13833) | 2 years ago |
| CODE_OF_CONDUCT.md | Update Code of Conduct to Contributor Covenant v2.1 (#19935) | last year |
| CONTRIBUTING.md | Fix the typos and grammar mistakes in CONTRIBUTING.md. (#27193) | 2 months ago |
| ISSUES.md | docs: replace torch.distributed.run by torchrun (#27528) | last month |
| LICENSE | Copyright (#8970) | 3 years ago |
| Makefile | Fix the check of models supporting FA/SDPA not run (#28202) | last week |
| README.md | Dev version | 3 weeks ago |
| README_es.md | Dev version | 3 weeks ago |
| README_hd.md | Dev version | 3 weeks ago |
| README_ja.md | Dev version | 3 weeks ago |
| README_ko.md | Dev version | 3 weeks ago |
| README_pt-br.md | Add madlad-400 MT models (#27471) | last month |
| README_ru.md | Add madlad-400 MT models (#27471) | last month |
| README_te.md | Add madlad-400 MT models (#27471) | last month |
| README_zh-hans.md | Dev version | 3 weeks ago |
| README_zh-hant.md | Dev version | 3 weeks ago |
| SECURITY.md | Added huggingface emoji instead of the markdown format (#27091) | 2 months ago |
| awesome-transformers.md | docs: add LLaMA-Efficient-Tuning to awesome-transformers (#25441) | 4 months ago |
| conftest.py | Bring back the PR `Refactor doctests + add CI` to `main` (#23271) | 7 months ago |
| hubconf.py | [`Styling`] stylify using ruff (#27144) | last month |
| pyproject.toml | [`Styling`] stylify using ruff (#27144) | last month |
| setup.py | Dev version | 3 weeks ago |

**About**

🤗 Transformers: State-of-the-art Machine Learning for Pytorch, TensorFlow, and JAX.

huggingface.co/transformers

python nlp machine-learning natural-language-processing deep-learning tensorflow pytorch transformer speech-recognition seq2seq flax pretrained-models language-models nlp-library hacktoberfest bert jax pytorch-transformers model-hub

Readme
Apache-2.0 license
Code of conduct
Security policy
Cite this repository ⌄
Activity
118k stars
1.1k watching
23.6k forks
Report repository

**Releases** 136

Patch release: v4.36.2 Latest
2 weeks ago

+ 135 releases

**Packages**

No packages published

**Used by** 109k

+ 109,602

**Contributors** 2,239

+ 2,225 contributors

**Languages**

● Python 99.5% ● Cuda 0.4%
● Shell 0.1% ● Dockerfile 0.0%
● C++ 0.0% ● C 0.0%

# 🤗 Transformers

build failing | license Apache-2.0 | website online | release v4.36.2 | Contributor Covenant v2.0 adopted | DOI 10.5281/zenodo.7391177

**English** | 简体中文 | 繁體中文 | 한국어 | Español | 日本語 | हिन्दी | Русский | Português | తెలుగు |

### State-of-the-art Machine Learning for JAX, PyTorch and TensorFlow

Part of the Hugging Face course!

🤗 Transformers provides thousands of pretrained models to perform tasks on different modalities such as text, vision, and audio.

These models can be applied on:

- 📝 Text, for tasks like text classification, information extraction, question answering, summarization, translation, and text generation, in over 100 languages.
- 🖼️ Images, for tasks like image classification, object detection, and segmentation.
- 🗣️ Audio, for tasks like speech recognition and audio classification.

Transformer models can also perform tasks on **several modalities combined**, such as table question answering, optical character recognition, information extraction from scanned documents, video classification, and visual question answering.

🤗 Transformers provides APIs to quickly download and use those pretrained models on a given text, fine-tune them on your own datasets and then share them with the community on our model hub. At the same time, each python module defining an architecture is fully standalone and can be modified to enable quick research experiments.

🤗 Transformers is backed by the three most popular deep learning libraries — Jax, PyTorch and TensorFlow — with a seamless integration between them. It's straightforward to train your models with one before loading them for inference with the other.

## Online demos

You can test most of our models directly on their pages from the model hub. We also offer private model hosting, versioning, & an inference API for public and private models.

Here are a few examples:

In Natural Language Processing:

- Masked word completion with BERT
- Name Entity Recognition with Electra
- Text generation with GPT-2
- Natural Language Inference with RoBERTa
- Summarization with BART
- Question answering with DistilBERT
- Translation with T5

In Computer Vision:

- Image classification with ViT
- Object Detection with DETR
- Semantic Segmentation with SegFormer
- Panoptic Segmentation with MaskFormer
- Depth Estimation with DPT
- Video Classification with VideoMAE
- Universal Segmentation with OneFormer

In Audio:

- Automatic Speech Recognition with Wav2Vec2
- Keyword Spotting with Wav2Vec2
- Audio Classification with Audio Spectrogram Transformer

In Multimodal tasks:

- Table Question Answering with TAPAS
- Visual Question Answering with ViLT
- Zero-shot Image Classification with CLIP
- Document Question Answering with LayoutLM
- Zero-shot Video Classification with X-CLIP

## 100 projects using Transformers

Transformers is more than a toolkit to use pretrained models: it's a community of projects built around it and the Hugging Face Hub. We want Transformers to enable developers, researchers, students, professors, engineers, and anyone else to build their dream projects.

In order to celebrate the 100,000 stars of transformers, we have decided to put the spotlight on the community, and we have created the awesome-transformers page which lists 100 incredible projects built in the vicinity of transformers.

If you own or use a project that you believe should be part of the list, please open a PR to add it!

## If you are looking for custom support from the Hugging Face team

Pssst...working with Transformers?

We have your back 💪💪💪

Learn more →

## Quick tour

To immediately use a model on a given input (text, image, audio, ...), we provide the `pipeline` API. Pipelines group together a pretrained model with the preprocessing that was used during that model's training. Here is how to quickly use a pipeline to classify positive versus negative texts:

```
>>> from transformers import pipeline

# Allocate a pipeline for sentiment-analysis
>>> classifier = pipeline('sentiment-analysis')
>>> classifier('We are very happy to introduce pipeline to the transformers repository.')
[{'label': 'POSITIVE', 'score': 0.9996980428695679}]
```

The second line of code downloads and caches the pretrained model used by the pipeline, while the third evaluates it on the given text. Here, the answer is "positive" with a confidence of 99.97%.

Many tasks have a pre-trained `pipeline` ready to go, in NLP but also in computer vision and speech. For example, we can easily extract detected objects in an image:

```
>>> import requests
>>> from PIL import Image
>>> from transformers import pipeline

# Download an image with cute cats
>>> url = "https://huggingface.co/datasets/huggingface/documentation-images/resolve/main/coco_sampl
>>> image_data = requests.get(url, stream=True).raw
>>> image = Image.open(image_data)

# Allocate a pipeline for object detection
>>> object_detector = pipeline('object-detection')
>>> object_detector(image)
[{'score': 0.9982201457023621,
  'label': 'remote',
  'box': {'xmin': 40, 'ymin': 70, 'xmax': 175, 'ymax': 117}},
 {'score': 0.9960021376609802,
  'label': 'remote',
  'box': {'xmin': 333, 'ymin': 72, 'xmax': 368, 'ymax': 187}},
 {'score': 0.9954745173454285,
  'label': 'couch',
  'box': {'xmin': 0, 'ymin': 1, 'xmax': 639, 'ymax': 473}},
 {'score': 0.9988006353378296,
  'label': 'cat',
  'box': {'xmin': 13, 'ymin': 52, 'xmax': 314, 'ymax': 470}},
 {'score': 0.9986783862113953,
  'label': 'cat',
  'box': {'xmin': 345, 'ymin': 23, 'xmax': 640, 'ymax': 368}}]
```

Here, we get a list of objects detected in the image, with a box surrounding the object and a confidence score. Here is the original image on the left, with the predictions displayed on the right:

You can learn more about the tasks supported by the `pipeline` API in this tutorial.

In addition to `pipeline`, to download and use any of the pretrained models on your given task, all it takes is three lines of code. Here is the PyTorch version:

```
>>> from transformers import AutoTokenizer, AutoModel

>>> tokenizer = AutoTokenizer.from_pretrained("bert-base-uncased")
>>> model = AutoModel.from_pretrained("bert-base-uncased")

>>> inputs = tokenizer("Hello world!", return_tensors="pt")
>>> outputs = model(**inputs)
```

And here is the equivalent code for TensorFlow:

```
>>> from transformers import AutoTokenizer, TFAutoModel

>>> tokenizer = AutoTokenizer.from_pretrained("bert-base-uncased")
>>> model = TFAutoModel.from_pretrained("bert-base-uncased")

>>> inputs = tokenizer("Hello world!", return_tensors="tf")
>>> outputs = model(**inputs)
```

The tokenizer is responsible for all the preprocessing the pretrained model expects and can be called directly on a single string (as in the above examples) or a list. It will output a dictionary that you can use in downstream code or simply directly pass to your model using the ** argument unpacking operator.

The model itself is a regular Pytorch `nn.Module` or a TensorFlow `tf.keras.Model` (depending on your backend) which you can use as usual. This tutorial explains how to integrate such a model into a classic PyTorch or TensorFlow training loop, or how to use our `Trainer` API to quickly fine-tune on a new dataset.

## Why should I use transformers?

1. Easy-to-use state-of-the-art models:
   - High performance on natural language understanding & generation, computer vision, and audio tasks.
   - Low barrier to entry for educators and practitioners.
   - Few user-facing abstractions with just three classes to learn.
   - A unified API for using all our pretrained models.
2. Lower compute costs, smaller carbon footprint:
   - Researchers can share trained models instead of always retraining.
   - Practitioners can reduce compute time and production costs.
   - Dozens of architectures with over 60,000 pretrained models across all modalities.
3. Choose the right framework for every part of a model's lifetime:
   - Train state-of-the-art models in 3 lines of code.
   - Move a single model between TF2.0/PyTorch/JAX frameworks at will.
   - Seamlessly pick the right framework for training, evaluation, and production.
4. Easily customize a model or an example to your needs:
   - We provide examples for each architecture to reproduce the results published by its original authors.
   - Model internals are exposed as consistently as possible.
   - Model files can be used independently of the library for quick experiments.

## Why shouldn't I use transformers?

- This library is not a modular toolbox of building blocks for neural nets. The code in the model files is not refactored with additional abstractions on purpose, so that researchers can quickly iterate on each of the models without diving into additional abstractions/files.
- The training API is not intended to work on any model but is optimized to work with the models provided by the library. For generic machine learning loops, you should use another library (possibly, Accelerate).
- While we strive to present as many use cases as possible, the scripts in our examples folder are just that: examples. It is expected that they won't work out-of-the-box on your specific problem and that you will be required to change a few lines of code to adapt them to your needs.

## Installation

### With pip

This repository is tested on Python 3.8+, Flax 0.4.1+, PyTorch 1.10+, and TensorFlow 2.6+.

You should install 🤗 Transformers in a virtual environment. If you're unfamiliar with Python virtual environments, check out the user guide.

First, create a virtual environment with the version of Python you're going to use and activate it.

Then, you will need to install at least one of Flax, PyTorch, or TensorFlow. Please refer to TensorFlow installation page, PyTorch installation page and/or Flax and Jax installation pages regarding the specific installation command for your platform.

When one of those backends has been installed, 🤗 Transformers can be installed using pip as follows:

```
pip install transformers
```

If you'd like to play with the examples or need the bleeding edge of the code and can't wait for a new release, you must install the library from source.

### With conda

Since Transformers version v4.0.0, we now have a conda channel: `huggingface`.

🤗 Transformers can be installed using conda as follows:

```
conda install -c huggingface transformers
```

Follow the installation pages of Flax, PyTorch or TensorFlow to see how to install them with conda.

> *NOTE:* On Windows, you may be prompted to activate Developer Mode in order to benefit from caching. If this is not an option for you, please let us know in this issue.

## Model architectures

**All the model checkpoints** provided by 🤗 Transformers are seamlessly integrated from the huggingface.co model hub, where they are uploaded directly by users and organizations.

Current number of checkpoints: models 451,025

🤗 Transformers currently provides the following architectures (see here for a high-level summary of each them):

1. **ALBERT** (from Google Research and the Toyota Technological Institute at Chicago) released with the paper ALBERT: A Lite BERT for Self-supervised Learning of Language Representations, by Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, Radu Soricut.
2. **ALIGN** (from Google Research) released with the paper Scaling Up Visual and Vision-Language Representation Learning With Noisy Text Supervision by Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc V. Le, Yunhsuan Sung, Zhen Li, Tom Duerig.
3. **AltCLIP** (from BAAI) released with the paper AltCLIP: Altering the Language Encoder in CLIP for Extended Language Capabilities by Chen, Zhongzhi and Liu, Guang and Zhang, Bo-Wen and Ye, Fulong and Yang, Qinghong and Wu, Ledell.
4. **Audio Spectrogram Transformer** (from MIT) released with the paper AST: Audio Spectrogram Transformer by Yuan Gong, Yu-An Chung, James Glass.
5. **Autoformer** (from Tsinghua University) released with the paper Autoformer: Decomposition Transformers with Auto-Correlation for Long-Term Series Forecasting by Haixu Wu, Jiehui Xu, Jianmin Wang, Mingsheng Long.
6. **Bark** (from Suno) released in the repository suno-ai/bark by Suno AI team.
7. **BART** (from Facebook) released with the paper BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension by Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer.
8. **BARThez** (from École polytechnique) released with the paper BARThez: a Skilled Pretrained French Sequence-to-Sequence Model by Moussa Kamal Eddine, Antoine J.-P. Tixier, Michalis Vazirgiannis.
9. **BARTpho** (from VinAI Research) released with the paper BARTpho: Pre-trained Sequence-to-Sequence Models for Vietnamese by Nguyen Luong Tran, Duong Minh Le and Dat Quoc Nguyen.
10. **BEiT** (from Microsoft) released with the paper BEiT: BERT Pre-Training of Image Transformers by Hangbo Bao, Li Dong, Furu Wei.
11. **BERT** (from Google) released with the paper BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding by Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova.
12. **BERT For Sequence Generation** (from Google) released with the paper Leveraging Pre-trained Checkpoints for Sequence Generation Tasks by Sascha Rothe, Shashi Narayan, Aliaksei Severyn.
13. **BERTweet** (from VinAI Research) released with the paper BERTweet: A pre-trained language model for English Tweets by Dat Quoc Nguyen, Thanh Vu and Anh Tuan Nguyen.
14. **BigBird-Pegasus** (from Google Research) released with the paper Big Bird: Transformers for Longer Sequences by Manzil Zaheer, Guru Guruganesh, Avinava Dubey, Joshua Ainslie, Chris Alberti, Santiago Ontanon, Philip Pham, Anirudh Ravula, Qifan Wang, Li Yang, Amr Ahmed.
15. **BigBird-RoBERTa** (from Google Research) released with the paper Big Bird: Transformers for Longer Sequences by Manzil Zaheer, Guru Guruganesh, Avinava Dubey, Joshua Ainslie, Chris Alberti, Santiago Ontanon, Philip Pham, Anirudh Ravula, Qifan Wang, Li Yang, Amr Ahmed.
16. **BioGpt** (from Microsoft Research AI4Science) released with the paper BioGPT: generative pre-trained transformer for biomedical text generation and mining by Renqian Luo, Liai Sun, Yingce Xia, Tao Qin, Sheng Zhang, Hoifung Poon and Tie-Yan Liu.
17. **BiT** (from Google AI) released with the paper Big Transfer (BiT): General Visual Representation Learning by Alexander Kolesnikov, Lucas Beyer, Xiaohua Zhai, Joan Puigcerver, Jessica Yung, Sylvain Gelly, Neil Houlsby.
18. **Blenderbot** (from Facebook) released with the paper Recipes for building an open-domain chatbot by Stephen Roller, Emily Dinan, Naman Goyal, Da Ju, Mary Williamson, Yinhan Liu, Jing Xu, Myle Ott, Kurt Shuster, Eric M. Smith, Y-Lan Boureau, Jason Weston.
19. **BlenderbotSmall** (from Facebook) released with the paper Recipes for building an open-domain chatbot by Stephen Roller, Emily Dinan, Naman Goyal, Da Ju, Mary Williamson, Yinhan Liu, Jing Xu, Myle Ott, Kurt Shuster, Eric M. Smith, Y-Lan Boureau, Jason Weston.
20. **BLIP** (from Salesforce) released with the paper BLIP: Bootstrapping Language-Image Pre-training for Unified Vision-Language Understanding and Generation by Junnan Li, Dongxu Li, Caiming Xiong, Steven Hoi.
21. **BLIP-2** (from Salesforce) released with the paper BLIP-2: Bootstrapping Language-Image Pre-training with Frozen Image Encoders and Large Language Models by Junnan Li, Dongxu Li, Silvio Savarese, Steven Hoi.
22. **BLOOM** (from BigScience workshop) released by the BigScience Workshop.
23. **BORT** (from Alexa) released with the paper Optimal Subarchitecture Extraction For BERT by Adrian de Wynter and Daniel J. Perry.
24. **BridgeTower** (from Harbin Institute of Technology/Microsoft Research Asia/Intel Labs) released with the paper BridgeTower: Building Bridges Between Encoders in Vision-Language Representation Learning by Xiao Xu, Chenfei Wu, Shachar Rosenman, Vasudev Lal, Wanxiang Che, Nan Duan.
25. **BROS** (from NAVER CLOVA) released with the paper BROS: A Pre-trained Language Model Focusing on Text and Layout for Better Key Information Extraction from Documents by Teakgyu Hong, Donghyun Kim, Mingi Ji, Wonseok Hwang, Daehyun Nam, Sungrae Park.
26. **ByT5** (from Google Research) released with the paper ByT5: Towards a token-free future with pre-trained byte-to-byte models by Linting Xue, Aditya Barua, Noah Constant, Rami Al-Rfou, Sharan Narang, Mihir Kale, Adam Roberts, Colin Raffel.
27. **CamemBERT** (from Inria/Facebook/Sorbonne) released with the paper CamemBERT: a Tasty French Language Model by Louis Martin*, Benjamin Muller*, Pedro Javier Ortiz Suárez*, Yoann Dupont, Laurent Romary, Éric Villemonte de la Clergerie, Djamé Seddah and Benoît Sagot.
28. **CANINE** (from Google Research) released with the paper CANINE: Pre-training an Efficient Tokenization-Free Encoder for Language Representation by Jonathan H. Clark, Dan Garrette, Iulia Turc, John Wieting.
29. **Chinese-CLIP** (from OFA-Sys) released with the paper Chinese CLIP: Contrastive Vision-Language Pretraining in Chinese by An Yang, Junshu Pan, Junyang Lin, Rui Men, Yichang Zhang, Jingren Zhou, Chang Zhou.
30. **CLAP** (from LAION-AI) released with the paper Large-scale Contrastive Language-Audio Pretraining with Feature Fusion and Keyword-to-Caption Augmentation by Yusong Wu, Ke Chen, Tianyu Zhang, Yuchen Hui, Taylor Berg-Kirkpatrick, Shlomo Dubnov.
31. **CLIP** (from OpenAI) released with the paper Learning Transferable Visual Models From Natural Language Supervision by Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, Ilya Sutskever.
32. **CLIPSeg** (from University of Göttingen) released with the paper Image Segmentation Using Text and Image Prompts by Timo Lüddecke and Alexander Ecker.
33. **CLVP** released with the paper Better speech synthesis through scaling by James Betker.
34. **CodeGen** (from Salesforce) released with the paper A Conversational Paradigm for Program Synthesis by Erik Nijkamp, Bo Pang, Hiroaki Hayashi, Lifu Tu, Huan Wang, Yingbo Zhou, Silvio Savarese, Caiming Xiong.
35. **CodeLlama** (from MetaAI) released with the paper Code Llama: Open Foundation Models for Code by Baptiste Rozière, Jonas Gehring, Fabian Gloeckle, Sten Sootla, Itai Gat, Xiaoqing Ellen Tan, Yossi Adi, Jingyu Liu, Tal Remez, Jérémy Rapin, Artyom Kozhevnikov, Ivan Evtimov, Joanna Bitton, Manish Bhatt, Cristian Canton Ferrer, Aaron Grattafiori, Wenhan Xiong, Alexandre Défossez, Jade Copet, Faisal Azhar, Hugo Touvron, Louis Martin, Nicolas Usunier, Thomas Scialom, Gabriel Synnaeve.
36. **Conditional DETR** (from Microsoft Research Asia) released with the paper Conditional DETR for Fast Training Convergence by Depu Meng, Xiaokang Chen, Zejia Fan, Gang Zeng, Houqiang Li, Yuhui Yuan, Lei Sun, Jingdong Wang.
37. **ConvBERT** (from YituTech) released with the paper ConvBERT: Improving BERT with Span-based Dynamic Convolution by Zihang Jiang, Weihao Yu, Daquan Zhou, Yunpeng Chen, Jiashi Feng, Shuicheng Yan.
38. **ConvNeXT** (from Facebook AI) released with the paper A ConvNet for the 2020s by Zhuang Liu, Hanzi Mao, Chao-Yuan Wu, Christoph Feichtenhofer, Trevor Darrell, Saining Xie.
39. **ConvNeXTV2** (from Facebook AI) released with the paper ConvNeXt V2: Co-designing and Scaling ConvNets with Masked Autoencoders by Sanghyun Woo, Shoubhik Debnath, Ronghang Hu, Xinlei Chen, Zhuang Liu, In So Kweon, Saining Xie.
40. **CPM** (from Tsinghua University) released with the paper CPM: A Large-scale Generative Chinese Pre-trained Language Model by Zhengyan Zhang, Xu Han, Hao Zhou, Pei Ke, Yuxian Gu, Deming Ye, Yujia Qin, Yusheng Su, Haozhe Ji, Jian Guan, Fanchao Qi, Xiaozhi Wang, Yanan Zheng, Guoyang Zeng, Huanqi Cao, Shengqi Chen, Daixuan Li, Zhenbo Sun, Zhiyuan Liu, Minlie Huang, Wentao Han, Jie Tang, Juanzi Li, Xiaoyan Zhu, Maosong Sun.
41. **CPM-Ant** (from OpenBMB) released by the OpenBMB.
42. **CTRL** (from Salesforce) released with the paper CTRL: A Conditional Transformer Language Model for Controllable Generation by Nitish Shirish Keskar*, Bryan McCann*, Lav R. Varshney, Caiming Xiong and Richard Socher.
43. **CvT** (from Microsoft) released with the paper CvT: Introducing Convolutions to Vision Transformers by Haiping Wu, Bin Xiao, Noel Codella, Mengchen Liu, Xiyang Dai, Lu Yuan, Lei Zhang.
44. **Data2Vec** (from Facebook) released with the paper Data2Vec: A General Framework for Self-supervised Learning in Speech, Vision and Language by Alexei Baevski, Wei-Ning Hsu, Qiantong Xu, Arun Babu, Jiatao Gu, Michael Auli.
45. **DeBERTa** (from Microsoft) released with the paper DeBERTa: Decoding-enhanced BERT with Disentangled Attention by Pengcheng He, Xiaodong Liu, Jianfeng Gao, Weizhu Chen.
46. **DeBERTa-v2** (from Microsoft) released with the paper DeBERTa: Decoding-enhanced BERT with Disentangled Attention by Pengcheng He, Xiaodong Liu, Jianfeng Gao, Weizhu Chen.
47. **Decision Transformer** (from Berkeley/Facebook/Google) released with the paper Decision Transformer: Reinforcement Learning via Sequence Modeling by Lili Chen, Kevin Lu, Aravind Rajeswaran, Kimin Lee, Aditya Grover, Michael Laskin, Pieter Abbeel, Aravind Srinivas, Igor Mordatch.
48. **Deformable DETR** (from SenseTime Research) released with the paper Deformable DETR: Deformable Transformers for End-to-End Object Detection by Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, Jifeng Dai.
49. **DeiT** (from Facebook) released with the paper Training data-efficient image transformers & distillation through attention by Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, Hervé Jégou.
50. **DePlot** (from Google AI) released with the paper DePlot: One-shot visual language reasoning by plot-to-table translation by Fangyu Liu, Julian Martin Eisenschlos, Francesco Piccinno, Syrine Krichene, Chenxi Pang, Kenton Lee, Mandar Joshi, Wenhu Chen, Nigel Collier, Yasemin Altun.
51. **DETA** (from The University of Texas at Austin) released with the paper NMS Strikes Back by Jeffrey Ouyang-Zhang, Jang Hyun Cho, Xingyi Zhou, Philipp Krähenbühl.
52. **DETR** (from Facebook) released with the paper End-to-End Object Detection with Transformers by Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, Sergey Zagoruyko.
53. **DialoGPT** (from Microsoft Research) released with the paper DialoGPT: Large-Scale Generative Pre-training for Conversational Response Generation by Yizhe Zhang, Siqi Sun, Michel Galley, Yen-Chun Chen, Chris Brockett, Xiang Gao, Jianfeng Gao, Jingjing Liu, Bill Dolan.
54. **DiNAT** (from SHI Labs) released with the paper Dilated Neighborhood Attention Transformer by Ali Hassani and Humphrey Shi.
55. **DINOv2** (from Meta AI) released with the paper DINOv2: Learning Robust Visual Features without Supervision by Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Mahmoud Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Hervé Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, Piotr Bojanowski.
56. **DistilBERT** (from HuggingFace), released together with the paper DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter by Victor Sanh, Lysandre Debut and Thomas Wolf. The same method has been applied to compress GPT2 into DistilGPT2, RoBERTa into DistilRoBERTa, Multilingual BERT into DistilmBERT and a German version of DistilBERT.
57. **DiT** (from Microsoft Research) released with the paper DiT: Self-supervised Pre-training for Document Image Transformer by Junlong Li, Yiheng Xu, Tengchao Lv, Lei Cui, Cha Zhang, Furu Wei.
58. **Donut** (from NAVER), released together with the paper OCR-free Document Understanding Transformer by Geewook Kim, Teakgyu Hong, Moonbin Yim, Jeongyeon Nam, Jinyoung Park, Jinyeong Yim, Wonseok Hwang, Sangdoo Yun, Dongyoon Han, Seunghyun Park.
59. **DPR** (from Facebook) released with the paper Dense Passage Retrieval for Open-Domain Question Answering by Vladimir Karpukhin, Barlas Oğuz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih.
60. **DPT** (from Intel Labs) released with the paper Vision Transformers for Dense Prediction by René Ranftl, Alexey Bochkovskiy, Vladlen Koltun.
61. **EfficientFormer** (from Snap Research) released with the paper EfficientFormer: Vision Transformers at MobileNetSpeed by Yanyu Li, Geng Yuan, Yang Wen, Ju Hu, Georgios Evangelidis, Sergey Tulyakov, Yanzhi Wang, Jian Ren.
62. **EfficientNet** (from Google Brain) released with the paper EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks by Mingxing Tan, Quoc V. Le.
63. **ELECTRA** (from Google Research/Stanford University) released with the paper ELECTRA: Pre-training text encoders as discriminators rather than generators by Kevin Clark, Minh-Thang Luong, Quoc V. Le, Christopher D. Manning.
64. **EnCodec** (from Meta AI) released with the paper High Fidelity Neural Audio Compression by Alexandre Défossez, Jade Copet, Gabriel Synnaeve, Yossi Adi.
65. **EncoderDecoder** (from Google Research) released with the paper Leveraging Pre-trained Checkpoints for Sequence Generation Tasks by Sascha Rothe, Shashi Narayan, Aliaksei Severyn.
66. **ERNIE** (from Baidu) released with the paper ERNIE: Enhanced Representation through Knowledge Integration by Yu Sun, Shuohuan Wang, Yukun Li, Shikun Feng, Xuyi Chen, Han Zhang, Xin Tian, Danxiang Zhu, Hao Tian, Hua Wu.
67. **ErnieM** (from Baidu) released with the paper ERNIE-M: Enhanced Multilingual Representation by Aligning Cross-lingual Semantics with Monolingual Corpora by Xuan Ouyang, Shuohuan Wang, Chao Pang, Yu Sun, Hao Tian, Hua Wu, Haifeng Wang.
68. **ESM** (from Meta AI) are transformer protein language models. **ESM-1b** was released with the paper Biological structure and function emerge from scaling unsupervised learning to 250 million protein sequences by Alexander Rives, Joshua Meier, Tom Sercu, Siddharth Goyal, Zeming Lin, Jason Liu, Demi Guo, Myle Ott, C. Lawrence Zitnick, Jerry Ma, and Rob Fergus. **ESM-1v** was released with the paper Language models enable zero-shot prediction of the effects of mutations on protein function by Joshua Meier, Roshan Rao, Robert Verkuil, Jason Liu, Tom Sercu and Alexander Rives. **ESM-2 and ESMFold** were released with the paper Language models of protein sequences at the scale of evolution enable accurate structure prediction by Zeming Lin, Halil Akin, Roshan Rao, Brian Hie, Zhongkai Zhu, Wenting Lu, Allan dos Santos Costa, Maryam Fazel-Zarandi, Tom Sercu, Sal Candido, Alexander Rives.
69. **Falcon** (from Technology Innovation Institute) by Almazrouei, Ebtesam and Alobeidli, Hamza and Alshamsi, Abdulaziz and Cappelli, Alessandro and Cojocaru, Ruxandra and Debbah, Merouane and Goffinet, Etienne and Heslow, Daniel and Launay, Julien and Malartic, Quentin and Noune, Badreddine and Pannier, Baptiste and Penedo, Guilherme.
70. **FLAN-T5** (from Google AI) released in the repository google-research/t5x by Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Eric Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, Albert Webson, Shixiang Shane Gu, Zhuyun Dai, Mirac Suzgun, Xinyun Chen, Aakanksha Chowdhery, Sharan Narang, Gaurav Mishra, Adams Yu, Vincent Zhao, Yanping Huang, Andrew Dai, Hongkun Yu, Slav Petrov, Ed H. Chi, Jeff Dean, Jacob Devlin, Adam Roberts, Denny Zhou, Quoc V. Le, and Jason Wei
71. **FLAN-UL2** (from Google AI) released in the repository google-research/t5x by Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Eric Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, Albert Webson, Shixiang Shane Gu, Zhuyun Dai, Mirac Suzgun, Xinyun Chen, Aakanksha Chowdhery, Sharan Narang, Gaurav Mishra, Adams Yu, Vincent Zhao, Yanping Huang, Andrew Dai, Hongkun Yu, Slav Petrov, Ed H. Chi, Jeff Dean, Jacob Devlin, Adam Roberts, Denny Zhou, Quoc V. Le, and Jason Wei
72. **FlauBERT** (from CNRS) released with the paper FlauBERT: Unsupervised Language Model Pre-training for French by Hang Le, Loïc Vial, Jibril Frej, Vincent Segonne, Maximin Coavoux, Benjamin Lecouteux, Alexandre Allauzen, Benoît Crabbé, Laurent Besacier, Didier Schwab.
73. **FLAVA** (from Facebook AI) released with the paper FLAVA: A Foundational Language And Vision Alignment Model by Amanpreet Singh, Ronghang Hu, Vedanuj Goswami, Guillaume Couairon, Wojciech Galuba, Marcus Rohrbach, and Douwe Kiela.
74. **FNet** (from Google Research) released with the paper FNet: Mixing Tokens with Fourier Transforms by James Lee-Thorp, Joshua Ainslie, Ilya Eckstein, Santiago Ontanon.
75. **FocalNet** (from Microsoft Research) released with the paper Focal Modulation Networks by Jianwei Yang, Chunyuan Li, Xiyang Dai, Lu Yuan, Jianfeng Gao.
76. **Funnel Transformer** (from CMU/Google Brain) released with the paper Funnel-Transformer: Filtering out Sequential Redundancy for Efficient Language Processing by Zihang Dai, Guokun Lai, Yiming Yang, Quoc V. Le.
77. **Fuyu** (from ADEPT) Rohan Bavishi, Erich Elsen, Curtis Hawthorne, Maxwell Nye, Augustus Odena, Arushi Somani, Sağnak Taşırlar. Released with the paper blog post
78. **GIT** (from Microsoft Research) released with the paper GIT: A Generative Image-to-text Transformer for Vision and Language by Jianfeng Wang, Zhengyuan Yang, Xiaowei Hu, Linjie Li, Kevin Lin, Zhe Gan, Zicheng Liu, Ce Liu, Lijuan Wang.
79. **GLPN** (from KAIST) released with the paper Global-Local Path Networks for Monocular Depth Estimation with Vertical CutDepth by Doyeon Kim, Woonghyun Ga, Pyungwhan Ahn, Donggyu Joo, Sehwan Chun, Junmo Kim.
80. **GPT** (from OpenAI) released with the paper Improving Language Understanding by Generative Pre-Training by Alec Radford, Karthik Narasimhan, Tim Salimans and Ilya Sutskever.
81. **GPT Neo** (from EleutherAI) released in the repository EleutherAI/gpt-neo by Sid Black, Stella Biderman, Leo Gao, Phil Wang and Connor Leahy.
82. **GPT NeoX** (from EleutherAI) released with the paper GPT-NeoX-20B: An Open-Source Autoregressive Language Model by Sid Black, Stella Biderman, Eric Hallahan, Quentin Anthony, Leo Gao, Laurence Golding, Horace He, Connor Leahy, Kyle McDonell, Jason Phang, Michael Pieler, USVSN Sai Prashanth, Shivanshu Purohit, Laria Reynolds, Jonathan Tow, Ben Wang, Samuel Weinbach
83. **GPT NeoX Japanese** (from ABEJA) released by Shinya Otani, Takayoshi Makabe, Anuj Arora, and Kyo Hattori.
84. **GPT-2** (from OpenAI) released with the paper Language Models are Unsupervised Multitask Learners by Alec Radford*, Jeffrey Wu*, Rewon Child, David Luan, Dario Amodei** and Ilya Sutskever**.
85. **GPT-J** (from EleutherAI) released in the repository kingoflolz/mesh-transformer-jax by Ben Wang and Aran Komatsuzaki.
86. **GPT-Sw3** (from AI-Sweden) released with the paper Lessons Learned from GPT-SW3: Building the First Large-Scale Generative Language Model for Swedish by Ariel Ekgren, Amaru Cuba Gyllensten, Evangelia Gogoulou, Alice Heiman, Severine Verlinden, Joey Öhman, Fredrik Carlsson, Magnus Sahlgren.
87. **GPTBigCode** (from BigCode) released with the paper SantaCoder: don't reach for the stars! by Loubna Ben Allal, Raymond Li, Denis Kocetkov, Chenghao Mou, Christopher Akiki, Carlos Munoz Ferrandis, Niklas Muennighoff, Mayank Mishra, Alex Gu, Manan Dey, Logesh Kumar Umapathi, Carolyn Jane Anderson, Yangtian Zi, Joel Lamy Poirier, Hailey Schoelkopf, Sergey Troshin, Dmitry Abulkhanov, Manuel Romero, Michael Lappert, Francesco De Toni, Bernardo García del Río, Qian Liu, Shamik Bose, Urvashi Bhattacharyya, Terry Yue Zhuo, Ian Yu, Paulo Villegas, Marco Zocca, Sourab Mangrulkar, David Lansky, Huu Nguyen, Danish Contractor, Luis Villa, Jia Li, Dzmitry Bahdanau, Yacine Jernite, Sean Hughes, Daniel Fried, Arjun Guha, Harm de Vries, Leandro von Werra.
88. **GPTSAN-japanese** released in the repository tanreinama/GPTSAN by Toshiyuki Sakamoto(tanreinama).
89. **Graphormer** (from Microsoft) released with the paper Do Transformers Really Perform Bad for Graph Representation? by Chengxuan Ying, Tianle Cai, Shengjie Luo, Shuxin Zheng, Guolin Ke, Di He, Yanming Shen, Tie-Yan Liu.
90. **GroupViT** (from UCSD, NVIDIA) released with the paper GroupViT: Semantic Segmentation Emerges from Text Supervision by Jiarui Xu, Shalini De Mello, Sifei Liu, Wonmin Byeon, Thomas Breuel, Jan Kautz, Xiaolong Wang.
91. **HerBERT** (from Allegro.pl, AGH University of Science and Technology) released with the paper KLEJ: Comprehensive Benchmark for Polish Language Understanding by Piotr Rybak, Robert Mroczkowski, Janusz Tracz, Ireneusz Gawlik.
92. **Hubert** (from Facebook) released with the paper HuBERT: Self-Supervised Speech Representation Learning by Masked Prediction of Hidden Units by Wei-Ning Hsu, Benjamin Bolte, Yao-Hung Hubert Tsai, Kushal Lakhotia, Ruslan Salakhutdinov, Abdelrahman Mohamed.
93. **I-BERT** (from Berkeley) released with the paper I-BERT: Integer-only BERT Quantization by Sehoon Kim, Amir Gholami, Zhewei Yao, Michael W. Mahoney, Kurt Keutzer.
94. **IDEFICS** (from HuggingFace) released with the paper OBELICS: An Open Web-Scale Filtered Dataset of Interleaved Image-Text Documents by Hugo Laurençon, Lucile Saulnier, Léo Tronchon, Stas Bekman, Amanpreet Singh, Anton Lozhkov, Thomas Wang, Siddharth Karamcheti, Alexander M. Rush, Douwe Kiela, Matthieu Cord, Victor Sanh.
95. **ImageGPT** (from OpenAI) released with the paper Generative Pretraining from Pixels by Mark Chen, Alec Radford, Rewon Child, Jeffrey Wu, Heewoo Jun, David Luan, Ilya Sutskever.
96. **Informer** (from Beihang University, UC Berkeley, Rutgers University, SEDD Company) released with the paper Informer: Beyond Efficient Transformer for Long Sequence Time-Series Forecasting by Haoyi Zhou, Shanghang Zhang, Jieqi Peng, Shuai Zhang, Jianxin Li, Hui Xiong, and Wancai Zhang.
97. **InstructBLIP** (from Salesforce) released with the paper InstructBLIP: Towards General-purpose Vision-Language Models with Instruction Tuning by Wenliang Dai, Junnan Li, Dongxu Li, Anthony Meng Huat Tiong, Junqi Zhao, Weisheng Wang, Boyang Li, Pascale Fung, Steven Hoi.
98. **Jukebox** (from OpenAI) released with the paper Jukebox: A Generative Model for Music by Prafulla Dhariwal, Heewoo Jun, Christine Payne, Jong Wook Kim, Alec Radford, Ilya Sutskever.
99. **KOSMOS-2** (from Microsoft Research Asia) released with the paper Kosmos-2: Grounding Multimodal Large Language Models to the World by Zhiliang Peng, Wenhui Wang, Li Dong, Yaru Hao, Shaohan Huang, Shuming Ma, Furu Wei.
100. **LayoutLM** (from Microsoft Research Asia) released with the paper LayoutLM: Pre-training of Text and Layout for Document Image Understanding by Yiheng Xu, Minghao Li, Lei Cui, Shaohan Huang, Furu Wei, Ming Zhou.
101. **LayoutLMv2** (from Microsoft Research Asia) released with the paper LayoutLMv2: Multi-modal Pre-training for Visually-Rich Document Understanding by Yang Xu, Yiheng Xu, Tengchao Lv, Lei Cui, Furu Wei, Guoxin Wang, Yijuan Lu, Dinei Florencio, Cha Zhang, Wanxiang Che, Min Zhang, Lidong Zhou.
102. **LayoutLMv3** (from Microsoft Research Asia) released with the paper LayoutLMv3: Pre-training for Document AI with Unified Text and Image Masking by Yupan Huang, Tengchao Lv, Lei Cui, Yutong Lu, Furu Wei.
103. **LayoutXLM** (from Microsoft Research Asia) released with the paper LayoutXLM: Multimodal Pre-training for Multilingual Visually-rich Document Understanding by Yiheng Xu, Tengchao Lv, Lei Cui, Guoxin Wang, Yijuan Lu, Dinei Florencio, Cha Zhang, Furu Wei.
104. **LED** (from AllenAI) released with the paper Longformer: The Long-Document Transformer by Iz Beltagy, Matthew E. Peters, Arman Cohan.
105. **LeViT** (from Meta AI) released with the paper LeViT: A Vision Transformer in ConvNet's Clothing for Faster Inference by Ben Graham, Alaaeldin El-Nouby, Hugo Touvron, Pierre Stock, Armand Joulin, Hervé Jégou, Matthijs Douze.
106. **LiLT** (from South China University of Technology) released with the paper LiLT: A Simple yet Effective Language-Independent Layout Transformer for Structured Document Understanding by Jiapeng Wang, Lianwen Jin, Kai Ding.
107. **LLaMA** (from The FAIR team of Meta AI) released with the paper LLaMA: Open and Efficient Foundation Language Models by Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, Guillaume Lample.
108. **Llama2** (from The FAIR team of Meta AI) released with the paper Llama2: Open Foundation and Fine-Tuned Chat Models by Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushka rMishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing EllenTan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, Thomas Scialom.
109. **LLaVa** (from Microsoft Research & University of Wisconsin-Madison) released with the paper Visual Instruction Tuning by Haotian Liu, Chunyuan Li, Yuheng Li and Yong Jae Lee.
110. **Longformer** (from AllenAI) released with the paper Longformer: The Long-Document Transformer by Iz Beltagy, Matthew E. Peters, Arman Cohan.