

AI 大模型开发工程师 之微调核心之Transformers库

讲师：李希沅

📖 目录

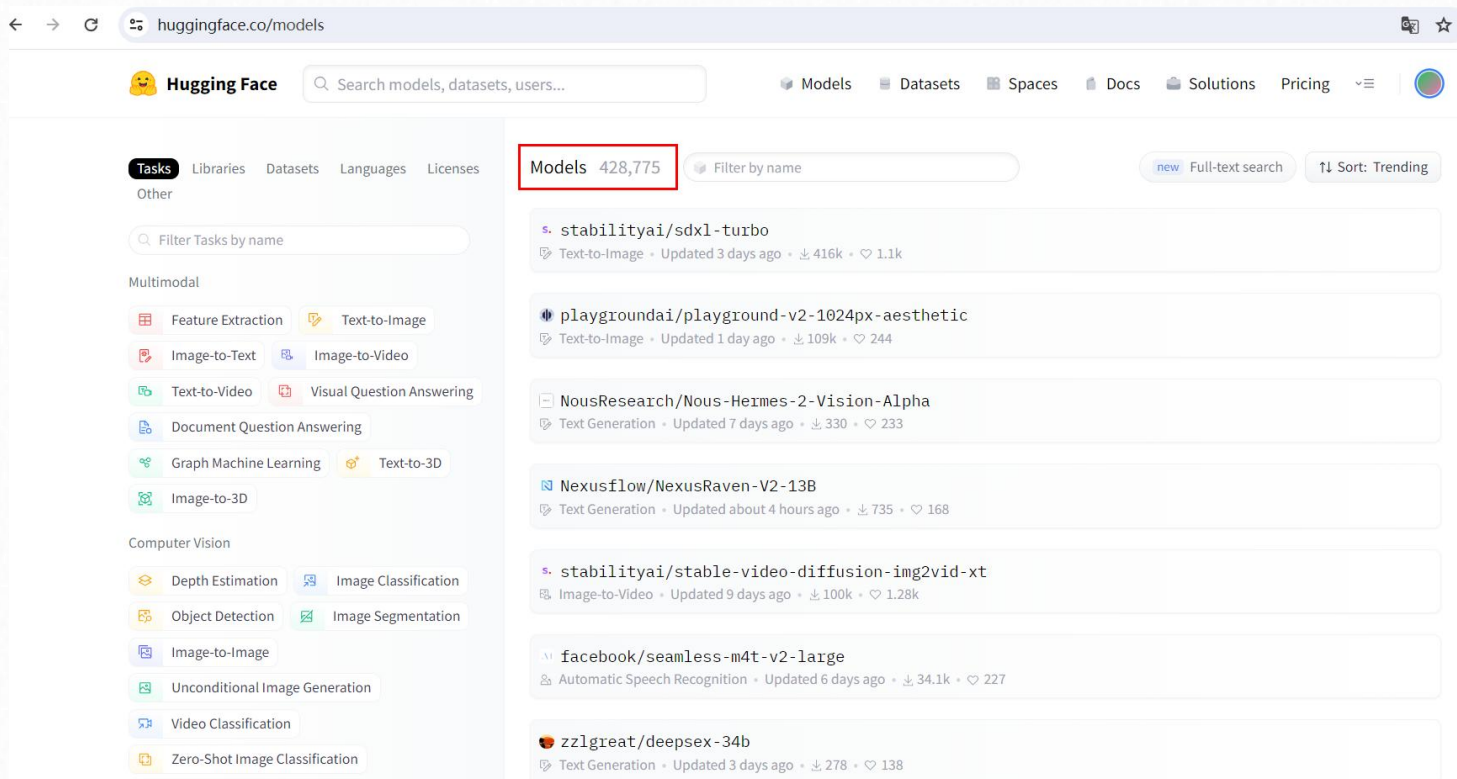
- | | | | |
|---|-----------------|---|------------|
| 1 | HuggingFace官网介绍 | 5 | Model组件 |
| 2 | 预训练编码流程初体验 | 6 | DataSets组件 |
| 3 | Pipeline组件 | 7 | Evaluate组件 |
| 4 | Tokenizer组件 | 8 | Trainer组件 |

1 HuggingFace官网介绍

01、HuggingFace官网介绍

官网地址：<https://huggingface.co/>

机器学习界的github



Models（模型）：包括各种处理CV和NLP等任务的模型，上面模型都是可以免费获得

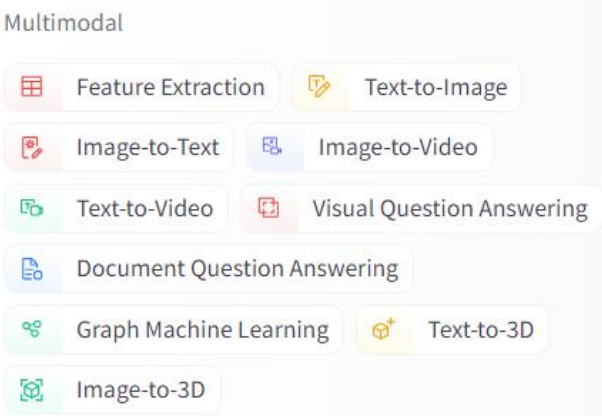
Datasets（数据集）：包括很多数据集

Spaces（分享空间）：包括社区空间下最新的一些有意思的分享，可以理解为huggingface朋友圈

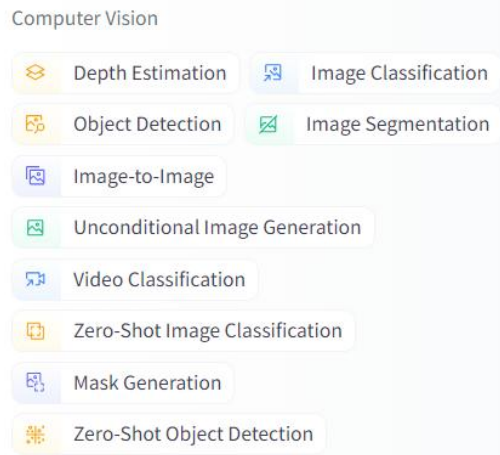
Docs（文档，各种模型算法文档）：包括各种模型算法等说明使用文档

Solutions（解决方案，体验等）：包括others

Pricing（计费）：提供专属服务



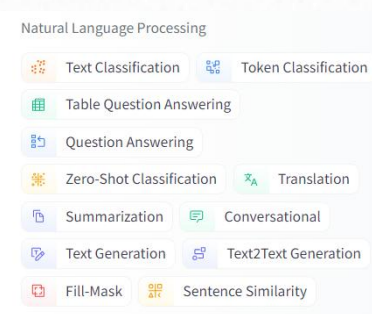
多模态



计算机视觉

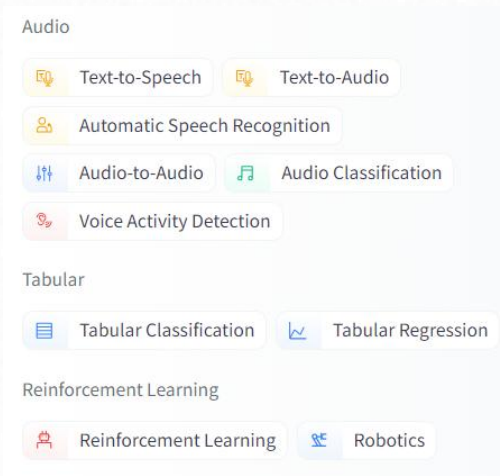
Feature Extraction (特征提取)、Text-to-Image (文本到图像)、Visual Question Answering (视觉问答)、Image2Text (图像到文本)、Document Question Answering (文档问答)

Computer Vision (计算机视觉任务) : 包括Image Classification (图像分类) , Image Segmentation (图像分割)、zero-Shot Image Classification (零样本图像分类)、Image-to-Image (图像到图像的任务)、Unconditional Image Generation (无条件图像生成)、Object Detection(目标检测)、Video Classification (视频分类)、Depth Estimation(深度估计, 估计拍摄者距离图像各处的距离)



自然语言处理

包括Translation (机器翻译)、Fill-Mask(填充掩码, 预测句子中被遮掩的词)、Token Classification (词分类)、Sentence Similarity (句子相似度)、Question Answering (问答系统) , Summarization (总结, 缩句)、Zero-Shot Classification (零样本分类)、Text Classification (文本分类)、Text2Text (文本到文本的生成)、Text Generation (文本生成)、Conversational (聊天)、Table Question Answer (表问答, 1.预测表格中被遮掩单词2.数字推理, 判断句子是否被表格数据支持)



音频, 表格处理、强化学习

Automatic Speech Recognition (语音识别)、Audio Classification (语音分类)、Text-to-Speech (文本到语音的生成)、Audio-to-Audio (语音到语音的生成)、Voice Activity Detection (声音检测、检测识别出需要的声音部分) Tabular Classification (表分类)、Tabular Regression (表回归) Reinforcement Learning (强化学习)、Robotics (机器人)

- **Hub**

Host Git-based models, datasets and Spaces on the Hugging Face Hub.

- **Hub Python Library**

Client library for the HF Hub: manage repositories from your Python runtime.

- **Inference API**

Experiment with over 200k models easily using our free Inference API.

- **Accelerate**

Easily train and use PyTorch models with multi-GPU, TPU, mixed-precision.

- **Tokenizers**

Fast tokenizers, optimized for both research and production.

- **Transformers**

State-of-the-art ML for Pytorch, TensorFlow, and JAX.

- **Datasets**

Access and share datasets for computer vision, audio, and NLP tasks.

- **Huggingface.js**

A collection of JS libraries to interact with Hugging Face, with TS types included.

- **Inference Endpoints**

Easily deploy models to production on dedicated, fully managed infrastructure.

- **Optimum**

Fast training and inference of HF Transformers with easy to use hardware optimization tools.

- **Evaluate**

Evaluate and report model performance easier and more standardized.

- **Diffusers**

State-of-the-art diffusion models for image and audio generation in PyTorch.

- **Gradio**

Build machine learning demos and other web apps, in just a few lines of Python.

- **Transformers.js**

Community library to run pretrained models from Transformers in your browser.

- **PEFT**

Parameter efficient finetuning methods for large models

- **AWS Trainium & Inferentia**

Train and Deploy Transformers & Diffusers with AWS Trainium and AWS Inferentia.

- **Tasks**

All things about ML tasks: demos, use cases, models, datasets, and more!

② 预训练编码流程初体验

01、案例场景

需求：基于目前的数据预训练一个分类模型，做到往这个模型里面输入一个评价，就知道是正面的评价还是负面的评价

label	review
1	距离川沙公路较近,但是公交指示不对,如果是“蔡陆线”的话,会非常麻烦. 建议用别的路线. 房间较为简单.
1	商务大床房, 房间很大, 床有2M宽, 整体感觉经济实惠不错!
1	早餐太差, 无论去多少人, 那边也不加食品的。酒店应该重视一下这个问题了。房间本身很好。
1	宾馆在小街道上, 不大好找, 但还好北京热心同胞很多~宾馆设施跟介绍的差不多, 房间很小, 确实挺小, 但加上低价位因素,
1	CBD中心, 周围没什么店铺, 说5星有点勉强. 不知道为什么卫生间没有电吹风
1	总的来说, 这样的酒店配这样的价格还算可以, 希望他赶快装修, 给我的客人留些好的印象
1	价格比比较不错的酒店。这次免费升级了, 感谢前台服务员。房子还好, 地毯是新的, 比上次的好些。早餐的人很多要早去些。

输入：昨晚在酒店里我睡得很好
模型的预测结果：好评

```
root@autodl-container-90c64fbe51-95b37e9b:~/transformers# pip show transformers
Name: transformers
Version: 4.35.2
Summary: State-of-the-art Machine Learning for JAX, PyTorch and TensorFlow
Home-page: https://github.com/huggingface/transformers
Author: The Hugging Face team (past and future) with the help of all our contributors (https://github.com/huggingface)
Author-email: transformers@huggingface.co
License: Apache 2.0 License
Location: /root/miniconda3/lib/python3.8/site-packages
Requires: numpy, filelock, regex, safetensors, packaging, requests, huggingface-hub, tqdm, pyyaml, tokenizers
Required-by:
```

数据地址: https://huggingface.co/datasets/dirtycomputer/ChnSentiCorp_htl_all
模型地址: <https://huggingface.co/hfl/rbt3>

步骤1：导入相关依赖

步骤2：获取数据集

步骤3：构建数据集

步骤4：划分数据集

步骤5：创建DataLoader

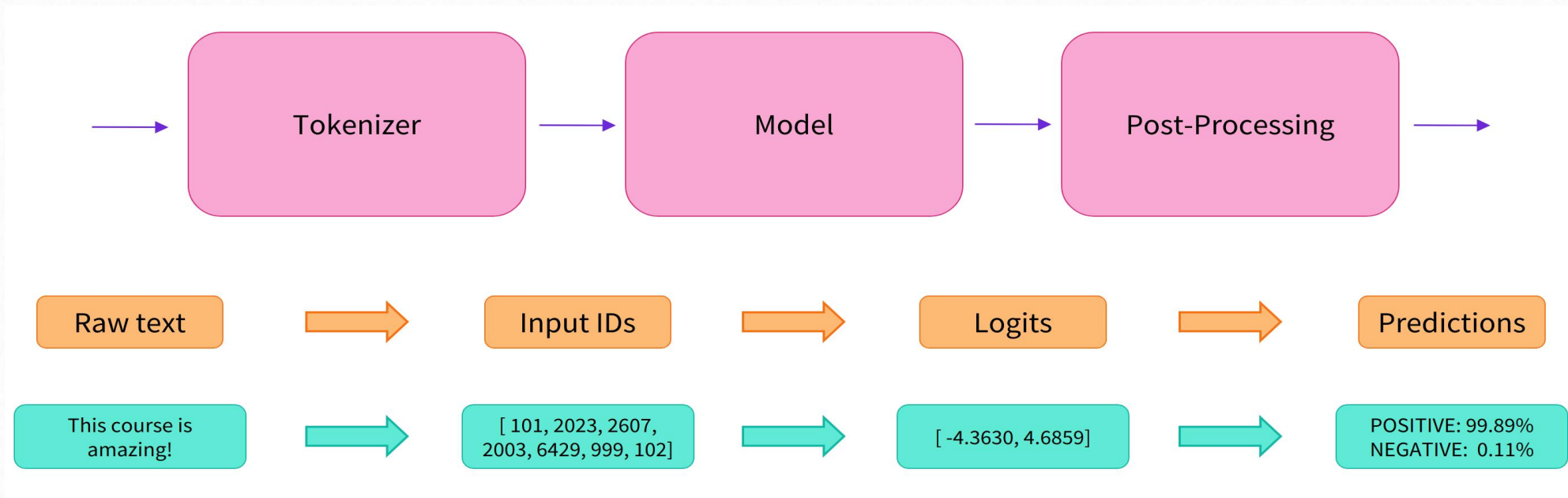
步骤6：创建模型及其优化器

步骤7：训练与验证

步骤8：模型预测

3 Pipeline组件

01、Pipeline组件



什么是Pipeline?

Pipeline如何使用?

Pipeline支持哪些任务类型?

Pipeline背后帮我们做了什么?

官网API地址: https://huggingface.co/docs/transformers/main_classes/pipelines

4 Tokenizer组件

01、Tokenizer组件

什么是Tokenizer?

Tokenizer如何使用?

Fast/Slow Tokenizer



Fast / Slow Tokenizer

FastTokenizer

基于Rust实现, 速度快

offsets_mapping, word_ids

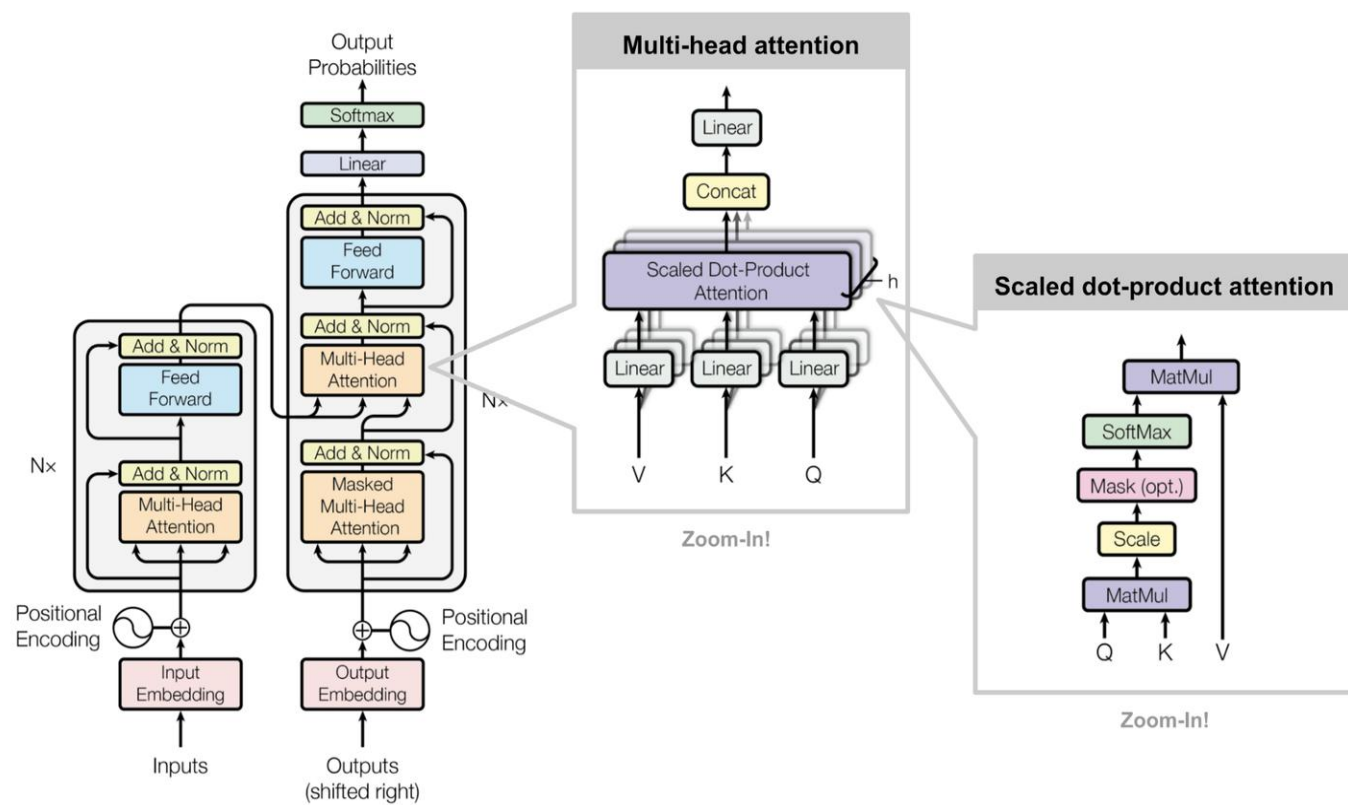
SlowTokenizer

基于Python实现, 速度慢

官网API地址: https://huggingface.co/docs/transformers/main_classes/tokenizer

5 Model组件

01、Transformer架构



Model类型介绍

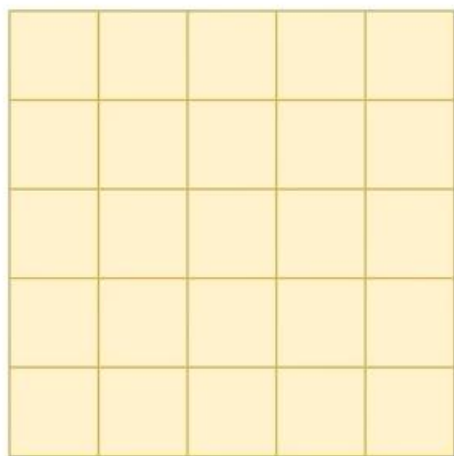
Model Head介绍

Model API调用

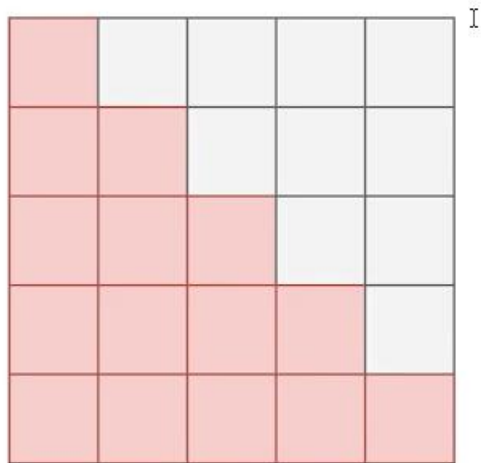
02、模型类型

目前基于Transformer的模型主要存在以下三种：

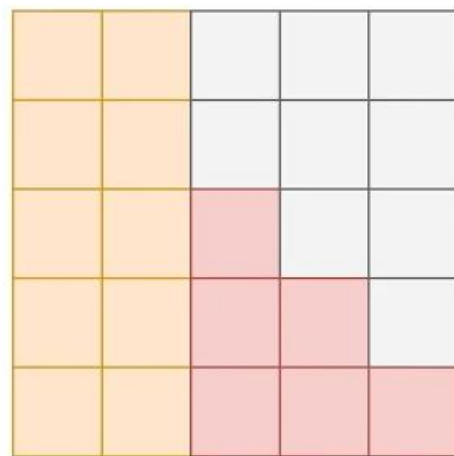
1. 仅仅包含Transformer的编码器模型（**自编码模型**）：，使用Encoder，可以从两个方向进行编码，拥有双向的注意力机制，即计算每一个词的特征时都看到完整上下文。常见仅仅存在编码器的预训练模型有：ALBERT,BERT,DistilBERT,RoBERTa等。经常被用于的任务：文本分类，命名实体识别，阅读理解等。
2. 仅仅存在Transformer的解码器模型：（**自回归模型**），使用Decoder，拥有单向的注意力机制，即计算每一个词的特征时都只能看到上文，无法看到下文。常见的预训练模型：GPT,GPT-2,GPT-3,Bloom,LLaMA等。经常被用于文本生成中。
3. 具有Transformers的编码器-解码器：（**序列到序列模型**），使用Encoder+Decoder，Encoder部分使用双向的注意力，Decoder部分使用单向注意力。常见的预训练模型为：BART，T5,mBART,GLM等。被用于文本摘要和机器翻译中。



编码器模型



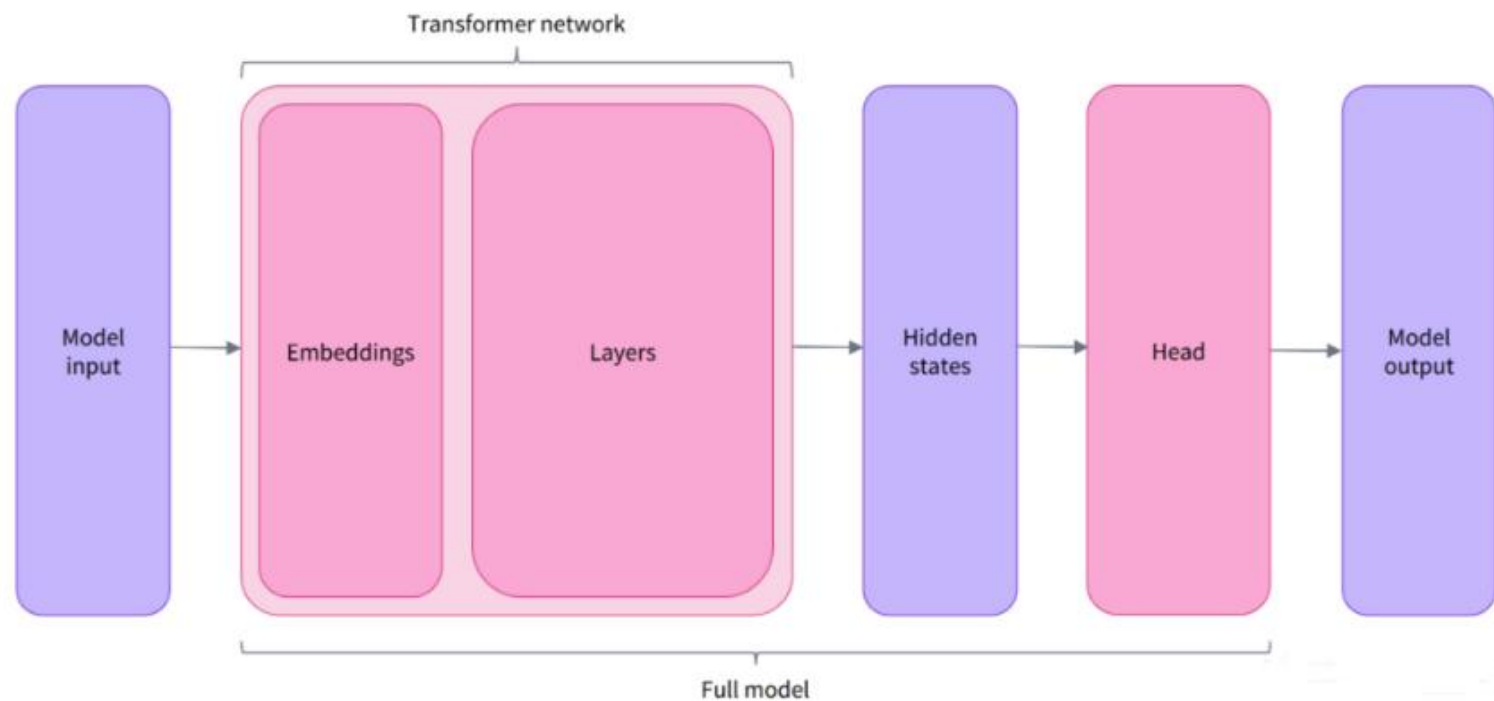
解码器模型



编码器解码器模型

03、Model head

- *model模型本身
- *ForCausalLM(解码器类型)
- *ForMaskedLM (编码器类型)
- *ForSeq2SeqLM
- *ForMultiplechoice
- *ForQuestionAnswering
- *ForSequenceClassification
- *ForTokenClassification
-



model head主要是将编码的表示结果进行映射，以解决不同类型的任务。

⑥ DataSets组件

01、DataSets组件

Datasets



😊 Datasets is a library for easily accessing and sharing datasets for Audio, Computer Vision, and Natural Language Processing (NLP) tasks.

Load a dataset in a single line of code, and use our powerful data processing methods to quickly get your dataset ready for training in a deep learning model. Backed by the Apache Arrow format, process large datasets with zero-copy reads without any memory constraints for optimal speed and efficiency. We also feature a deep integration with the Hugging Face Hub, allowing you to easily load and share a dataset with the wider machine learning community.

官网API: <https://huggingface.co/docs/datasets/index>

在线加载数据集

数据集选取,过滤,映射

DataCollator

查看数据集

数据集保存与加载


改造预训练代码

数据集划分


加载本地数据集

7 Evaluate组件

01、Evaluate组件




Evaluate

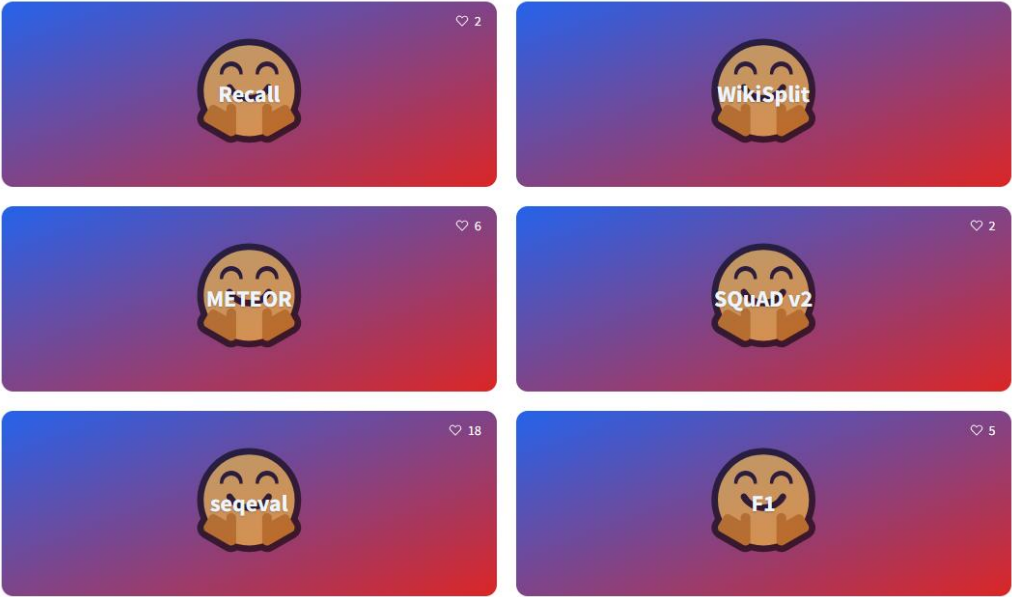


Evaluate

A library for easily evaluating machine learning models and datasets.

With a single line of code, you get access to dozens of evaluation methods for different domains (NLP, Computer Vision, Reinforcement Learning, and more!). Be it on your local machine or in a distributed training setup, you can evaluate your models in a consistent and reproducible way!

Visit the  Evaluate [organization](#) for a full list of available metrics. Each metric has a dedicated Space with an interactive demo for how to use the metric, and a documentation card detailing the metrics limitations and usage.



官网的API: <https://huggingface.co/docs/evaluate/index>

测试小案例: <https://huggingface.co/evaluate-metric>

Evaluate组件介绍

Evaluate API用法

改造预训练代码

- **Tokenizers**
Fast tokenizers, optimized for both research and production.
- **Datasets-server**
API to access the contents, metadata and basic statistics of all Hugging Face Hub datasets.
- **tim**
State-of-the-art computer vision models, layers, optimizers, training/evaluation, and utilities.
- **AutoTrain**
AutoTrain API and UI

- **Evaluate**
Evaluate and report model performance easier and more standardized.
- **TRL**
Train transformer language models with reinforcement learning.
- **Safetensors**
Simple, safe way to store and distribute neural networks weights safely and quickly.
- **Text Embeddings Inference**
Toolkit to serve Text Embedding Models.

- **Tasks**
All things about ML tasks: demos, use cases, models, datasets, and more!
- **Amazon SageMaker**
Train and Deploy Transformer models with Amazon SageMaker and Hugging Face DLCs.
- **Text Generation Inference**
Toolkit to serve Large Language Models.

任务类型: <https://huggingface.co/tasks>

8 Trainer组件

01、Trainer组件

Trainer模块主要包含两部分的内容：TrainingArguments与Trainer，前者用于训练参数的设置，后者用于创建真正的训练器，进行训练、评估预测等实际操作。

TrainingArguments

TrainingArguments中可以配置整个训练过程中使用的参数，默认版本是包含90个参数，涉及模型存储、模型优化、训练日志、GPU使用、模型精度、分布式训练等多方面的配置内容。

Trainer

Trainer中配置具体的训练用到的内容，包括模型、训练参数、训练集、验证集、分词器、评估函数等内容。

```
from transformers import TrainingArguments, Trainer
# 创建TrainingArguments
training_args = TrainingArguments(...)
# 创建Trainer
trainer = Trainer(..., args=training_args, ...)
# 模型训练
trainer.train()
# 模型评估
trainer.evaluate()
# 模型预测
trainer.predict()
```

Trainer组件介绍

基于Trainer优化预训练代码

官网API: https://huggingface.co/docs/transformers/main_classes/trainer

关注视频号：玄姐谈AGI
助力数字化人才提升 AIGC 能力



玄姐谈 AGI 



扫一扫二维码，关注我的视频号