

01

基础知识与环境安装

- (1) 自然语言处理任务
- (2) Transformers介绍
- (3) Transformers相关环境安装
- (4) 两行代码的QA实例

Transformers基础知识

常见自然语言处理任务

情感分析 (sentiment-analysis) : 对给定的文本分析其情感极性

文本生成 (text-generation) : 根据给定的文本进行生成

命名实体识别 (ner) : 标记句子中的实体

阅读理解 (question-answering) : 给定上下文与问题, 从上下文中抽取答案

掩码填充 (fill-mask) : 填充给定文本中的掩码词

文本摘要 (summarization) : 生成一段长文本的摘要

机器翻译 (translation) : 将文本翻译成另一种语言

特征提取 (feature-extraction) : 生成给定文本的张量表示

对话机器人 (conversational) : 根据用户输入文本, 产生回应, 与用户对话

自然语言处理的几个阶段

- 第一阶段：统计模型 + 数据（特征工程）
 - 决策树、SVM、HMM、CRF、TF-IDF、BOW
- 第二阶段：神经网络 + 数据
 - Linear、CNN、RNN、GRU、LSTM、Transformer、Word2vec、Glove
- 第三阶段：神经网络 + 预训练模型 + （少量）数据
 - GPT、BERT、RoBERTa、ALBERT、BART、T5
- 第四阶段：神经网络 + 更大的预训练模型 + Prompt
 - ChatGPT、Bloom、LLaMA、Alpaca、Vicuna、MOSS、文心一言、通义千问、星火

Transformers基础知识

Transformers简单介绍

- 官方网址: <https://huggingface.co/>
- HuggingFace出品, 当下最热、最常使用的自然语言处理工具包之一, 不夸张的说甚至没有之一
- 实现了大量的基于Transformer架构的主流预训练模型, 不局限于自然语言处理模型, 还包括图像、音频以及多模态的模型
- 提供了海量的预训练模型与数据集, 同时支持用户自行传, 社区完善, 文档全面, 三两行代码便可快速实现模型训练推理, 上手简单

一句话总结: 学就对了!

Transformers及相关库

- Transformers: 核心库, 模型加载、模型训练、流水线等
- Tokenizer: 分词器, 对数据进行预处理, 文本到token序列的互相转换
- Datasets: 数据集库, 提供了数据集的加载、处理等方法
- Evaluate: 评估函数, 提供各种评价指标的计算函数
- PEFT: 高效微调模型的库, 提供了几种高效微调的方法, 小参数量撬动大模型
- Accelerate: 分布式训练, 提供了分布式训练解决方案, 包括大模型的加载与推理解决方案
- Optimum: 优化加速库, 支持多种后端, 如Onnxruntime、OpenVino等
- Gradio: 可视化部署库, 几行代码快速实现基于Web交互的算法演示系统

Transformers环境安装

前置环境安装——python

- **miniconda 安装**

- 下载地址: <https://mirrors.tuna.tsinghua.edu.cn/anaconda/miniconda/>
- 如果C盘有空间, 最好安装在C盘, 且安装目录中不能有中文
- 勾选将其添加到PATH

- **conda环境创建**

- 命令: `conda create -n transformers python=3.9`
- 明确指定版本, 否则可能会因版本过高导致有包装不上

- **pypi配置国内源**

- 清华源: <https://mirrors.tuna.tsinghua.edu.cn/help/pypi/>

Transformers环境安装

前置环境安装—pytorch

- **pytorch安装**

- 官方地址: <https://pytorch.org/>
- 在一个单独的环境中, 能使用pip就尽量使用pip, 实在有问题的情况, 例如没有合适的编译好的系统版本的安装包, 再使用conda进行安装, 不要来回混淆
- 30XX、40XX显卡, 要安装cu11以上的版本, 否则无法运行

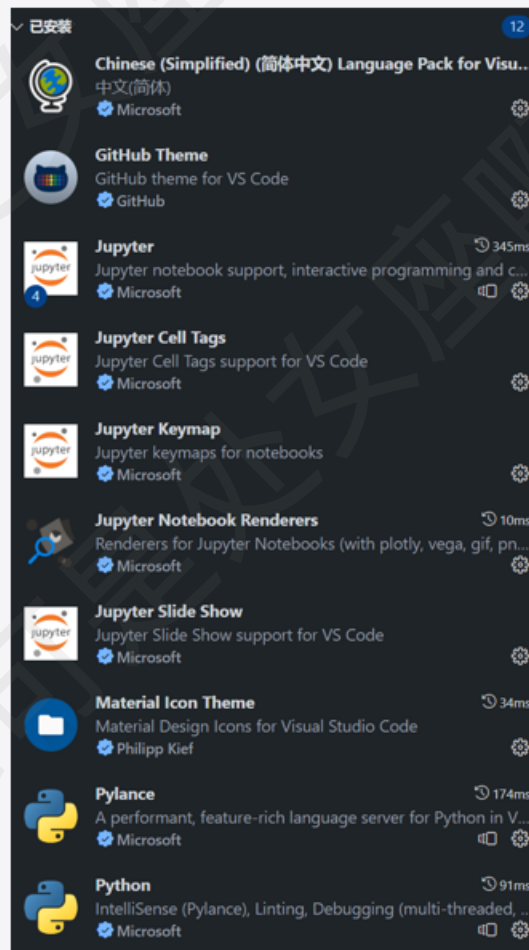
- **CUDA是否要安装**

- 如果只需要训练、简单推理, 则无需单独安装CUDA, 直接安装pytorch
- 如果有部署需求, 例如导出TensorRT模型, 则需要安装CUDA

Transformers环境安装

前置环境安装—vscode

- VS Code 安装
 - 官方地址: <https://code.visualstudio.com/download>
- 插件安装
 - Python (代码编写)
 - remote ssh (连接服务器)
 - Chinese Language Pack (简体中文包)
- 终端设置 (非常重要! 非常重要! 非常重要!)
 - 选择默认配置文件: cmd.exe



Transformers环境安装

Transformers安装

- **安装命令**

- `pip install transformers datasets evaluate peft accelerate gradio optimum sentencepiece`
- `pip install jupyterlab scikit-learn pandas matplotlib tensorboard nltk rouge`

- **hosts修改 (暂时用不上了)**

- `185.199.108.133 raw.githubusercontent.com`
- `185.199.109.133 raw.githubusercontent.com`
- `185.199.110.133 raw.githubusercontent.com`
- `185.199.111.133 raw.githubusercontent.com`
- `2606:50c0:8000::154 raw.githubusercontent.com`
- `2606:50c0:8001::154 raw.githubusercontent.com`
- `2606:50c0:8002::154 raw.githubusercontent.com`
- `2606:50c0:8003::154 raw.githubusercontent.com`



Transformers极简实例

三行代码，启动NLP应用

样例1：文本分类

```
# 导入gradio
import gradio as gr
# 导入transformers相关包
from transformers import *
# 通过Interface加载pipeline并启动文本分类服务
gr.Interface.from_pipeline(pipeline("text-classification", model="uer/roberta-base-finetuned-dianping-chinese")).launch()
```

样例2：阅读理解

```
# 导入gradio
import gradio as gr
# 导入transformers相关包
from transformers import *
# 通过Interface加载pipeline并启动阅读理解服务
gr.Interface.from_pipeline(pipeline("question-answering", model="uer/roberta-base-chinese-extractive-qa")).launch()
```