|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 类别 | 正则表达式 | TE-IDF | BERT | 大语言模型 |
| 优点 | 1. 效率非常高，因为只是字符串的匹配，不涉及其他数据训练； 2. 分类依据自己定义，非常清晰； 3. 可以在CPU上运行，不需占用GPU | 1. 识别准确度相比正则表达式有很大提升，运用了机器学习的算法，能够简单的理解语义； 2. 可以在CPU上运行。模型比较轻便 | 1. 能够结合上下文理解语义，分类的准确率更高，对于一词多义也能够区分； 2. 训练一次就能够用在其他项目上进行语义的分析 | 1. 处理语义的能力最强大，对于复杂的句子也能够清楚的分类； 2. 不需要大量的数据输入，直接输入想要分类的语句就能够得到答案 |
| 缺点 | 1. 新增类别只能通过人工手动增加，可能会存在人工漏掉的情况，比较麻烦； 2. 只能通过从现有的类别中查找识别，如果不在范围内，就会被归类为Other 3. 没有办法根据上下文来理解含义，能匹配上就识别成功，不能匹配上就识别失败 | 1. 用到了中文分词，这个分词的质量不好的话会导致识别出现问题，比较依赖分词； 2. 不能够结合上下文对词语进行准确的理解，对于一词多义的情况没有较好的分辨能力 | 1. 模型比较大，训练比较耗时； 2. 首次训练的时候会生成一个权重文件，在后面训练时需要作为输入； 3. 分类的依据不像正则表达式那样可以直接查看和修改 | 1. 本地的大模型部署需要很大的内存； 2. 速度比较慢 |