

用户使用手册

1. 项目介绍

本项目是一个基于深度学习的中文文本分类系统，支持多种模型架构，如 **TextCNN**、**TextRNN** 和 **FastText**。用户可以通过 Web 应用界面输入中文文本，系统会自动将文本分类到相应的类别中。数据集来源于 **THUCNews**，包括多个新闻类型，以确保分类模型的广泛性和准确性。

2. 环境配置

- **Python**: 确保已安装 Python 3.6 及以上版本。
- **依赖包**: 在项目根目录下运行以下命令，安装所有必要的依赖包：
`pip install -r requirements.txt`
- **PyTorch**: 用于深度学习模型的训练和预测。
- **Flask**: 用于提供 Web 界面和服务接口。
- **TensorBoard**: 用于可视化训练过程中的损失和准确率。

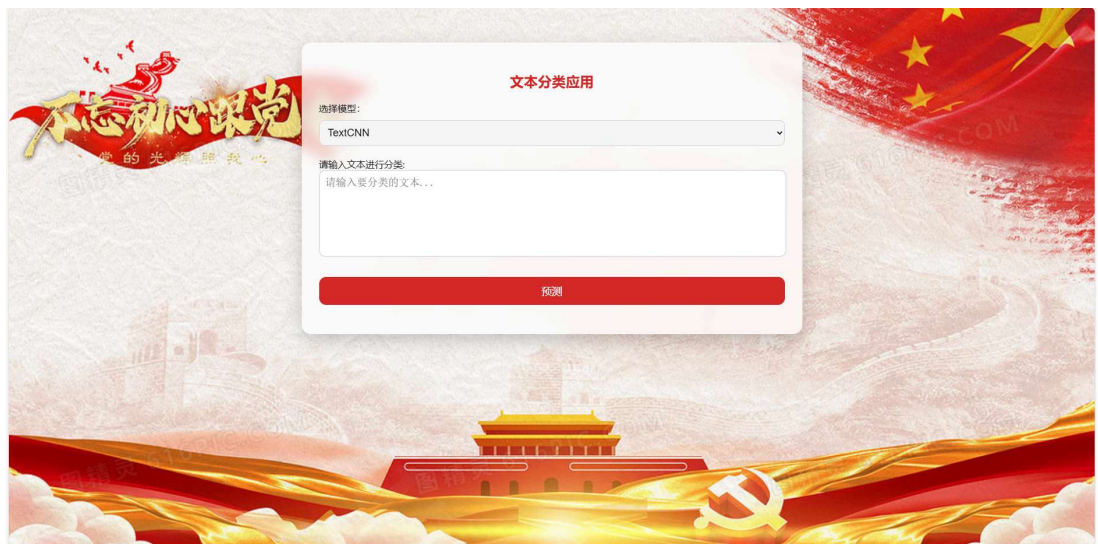
3. 文件结构介绍

- **models 文件夹**: 包含各种模型的实现。
 - **TextCNN.py**: 卷积神经网络模型文件。
 - **TextRNN.py**: 循环神经网络模型文件。
- **THUCNews/data 文件夹**: 存放数据集文件。
 - **train.txt**、**dev.txt**、**test.txt** **: 分别是训练集、验证集和测试集。
- **saved_dict 文件夹**: 存放训练后的模型参数。
- **app.py**: 用于启动 Flask Web 服务的主程序文件。
- **templates/index.html**: 用户界面 HTML 模板。

4. 使用步骤

4.4 使用截图

- **Web 界面截图**: 展示在浏览器中访问 Flask 应用后的界面，以及用户输入文本进行分类的操作。



- **分类结果展示截图：**展示系统对输入文本的分类结果。

选择模型：

TextCNN

请输入文本进行分类:

习近平离京赴利马出席亚太经合组织会议

预测

预测结果：

预测标签: politics

预测概率: 99.76646900177002%

所有标签的概率:

finance:	0.03840888966806233%
realty:	0.003146034214296378%
stocks:	0.03804937587119639%
education:	0.008402900857618079%
science:	0.008227676153182983%
society:	0.10366846108809114%
politics:	99.76646900177002%
sports:	0.026023216196335852%
game:	0.000697422910889145%
entertainment:	0.006911589298397303%

请输入文本进行分类:

习近平离京赴利马出席亚太经合组织会议

预测

预测结果:

预测标签: politics

预测概率: 99.79203343391418%

所有标签的概率:

finance: 0.02587867493275553%

realty: 0.0076014912337996066%

stocks: 0.11959138792008162%

education: 0.004122416066820733%

science: 0.04384350904729217%

society: 0.0037483725463971496%

politics: 99.79203343391418%

sports: 0.0006752635727025336%

game: 0.0013926196515967604%

entertainment: 0.0011086648555647116%

请输入文本进行分类:

中字头，多股涨停！融资客大举加仓

预测

预测结果:

预测标签: stocks

预测概率: 98.32777380943298%

所有标签的概率:

finance: 1.472549233585596%

realty: 0.018910221115220338%

stocks: 98.32777380943298%

education: 6.800495384595706e-05%

science: 0.1669165794737637%

society: 9.704846206659568e-05%

politics: 0.012166430678917095%

sports: 0.00016719242239560117%

game: 0.0013245315130916424%

entertainment: 2.479899308127642e-05%

TextRNN

请输入文本进行分类:
中字头，多股涨停！融资客大举加仓

预测

预测结果:

预测标签: stocks

预测概率: 98.32777380943298%

所有标签的概率:

finance: 1.472549233585596%

reality: 0.018910221115220338%

stocks: 98.32777380943298%

education: 6.800495384595706e-05%

science: 0.1669165794737637%

society: 9.704846206659568e-05%

politics: 0.012166430678917095%

sports: 0.00016719242239560117%

game: 0.0013245315130916424%

entertainment: 2.479899308127642e-05%

4.1 模型训练

1. 进入项目根目录。
2. 执行 `run.py` 文件，选择合适的模型进行训练。

```
python run.py --model TextCNN --embedding pre_trained --word False
```

 - `--model`：选择要训练的模型类型（如 `TextCNN`）。
 - `--embedding`：选择使用预训练词嵌入还是随机初始化。
 - `--word`：选择按字（`False`）还是按词（`True`）进行分词。

4.2 启动 Web 应用

1. 训练完成后，运行 `app.py` 启动 Flask Web 服务。

```
python app.py
```
2. 在浏览器中访问 `http://localhost:5000`，进入分类系统的用户界面。

4.3 使用 Web 界面进行文本分类

1. 在输入框中输入一段中文文本。
2. 点击 "分类" 按钮。
3. 系统将调用后端模型对文本进行分类，并在页面上显示分类结果。

5. 功能介绍

- **模型选择与训练**: 用户可以通过修改 `run.py` 中的参数, 选择不同的模型 (如 TextCNN、TextRNN) 进行训练。每种模型的优劣在不同场景中各有特点, 用户可根据具体应用场景选择合适的模型。
- **Web 界面交互**: 通过简单的前端界面, 用户可以输入文本, 并获得模型的分类结果, 方便易用。
- **日志与可视化**: 通过 TensorBoard 可视化工具, 用户可以在训练过程中跟踪模型的损失和准确率变化。

9. 结语

本用户手册旨在帮助用户理解如何使用该中文文本分类系统。通过简便的 Web 界面, 用户无需掌握深度学习的技术细节即可进行文本分类。希望本手册能对您的使用提供帮助。