



上海交通大学  
SHANGHAI JIAO TONG UNIVERSITY

## 2025 《人工智能导论》大作业

### NIS4307 Rumor Detector

任务名称:	Rumor Detector
完成组号:	第 1 组
小组成员:	马悦钊 李卓恒 刘梓芑 聂鸣涛
完成时间:	2025 年 5 月 25 日

## 目 录

第 1 章 任务目标 .....	2
第 2 章 具体内容 .....	3
2.1 实施方案.....	3
2.2 核心代码分析.....	3
2.3 测试结果分析.....	4
第 3 章 工作总结 .....	5
3.1 收获与心得.....	5
3.2 遇到问题及解决思路.....	5
第 4 章 课程建议 .....	6

## 第 1 章 任务目标

本次课程设计的任务是基于谣言检测数据集，构建一个检测模型。该模型可以对数据集集中的推文进行谣言检测与识别。要求如下：

- 数据集：使用给定的谣言检测数据集，数据集包含推文文本和标签（谣言或非谣言）。
- 训练模型：使用逻辑回归或 GRU 等深度学习模型进行谣言检测，实现二分类任务，用 0 代表非谣言、1 代表谣言
- 泛化能力：模型应具有较好的泛化能力，能够适应不同类型的谣言检测任务。
- 评估指标：分类准确率、运行时间等
- 结果可视化：对模型训练结果进行可视化展示。

我们需要在接口类文件 `classify.py` 中实现接口类 `RumourDetectClass`，该类对外提供一个接口函数 `classify`，该函数接收一条字符串作为输入，输出一个 `int` 值作为对应的预测类别。该类共包含以下方法：

- `__init__(self)`: 初始化类

## 第2章 具体内容

### 2.1 实施方案

### 2.2 核心代码分析

```
import re
from train_gru import *

class RumourDetectClass:
    def __init__(self):
        # 加载词表和模型参数
        self.vocab = build_vocab(pd.read_csv('../dataset/split/train
.csv')['text'])
        self.model = BiGRU(len(self.vocab), EMBEDDING_DIM,
HIDDEN_DIM).to(DEVICE)
        self.model.load_state_dict(torch.load('../Output/bigru.pt',
map_location=DEVICE))
        self.model.eval()

    def preprocess(self, text):
        # 文本预处理（与训练时一致）
        text = re.sub(r'^\w\s', '', text.lower())
        return text

    def classify(self, text: str) -> int:
        # 预测流程
        text = self.preprocess(text)
        ids = encode(text, self.vocab)
        x = torch.tensor([ids], dtype=torch.long).to(DEVICE)
        with torch.no_grad():
            logits = self.model(x)
            pred = (torch.sigmoid(logits) > 0.5).float().item()
        return int(pred)
```

## 2.3 测试结果分析

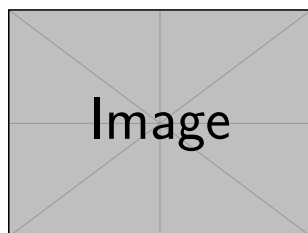


图 2.1 示例图片

## 第 3 章 工作总结

### 3.1 收获与心得

### 3.2 遇到问题及解决思路

## 第 4 章 课程建议