**项目文档说明**

**项目目标**

实现一个**两阶段弱监督时序动作定位（WS-TAL）系统**，仅使用视频级标签训练模型，实现：

1. **动作片段识别**（定位动作发生的时间区间）
2. **动作分类**（识别动作类别）
3. **边界优化**（精确调整动作边界）

**核心思路与模块**

**阶段1：S\_Model**

**目标**：生成初始动作候选片段（proposals）和可靠性评估

**核心组件**：

* Reliable\_Memory：类原型记忆库
* Reliabilty\_Aware\_Block：可靠性感知特征增强模块
* Encoder：特征编码器

**工作流程**：

1. **特征增强**：通过可靠性感知块增强视频特征。
2. **类别激活**：生成帧级类别激活序列。
3. **对比学习**：
   * 使用记忆库存储类原型特征。
   * 拉近动作特征与类原型的距离。
   * 推远动作特征与背景特征的距离。
4. **损失函数**：
   * 视频级分类损失。
   * 帧级动作/背景分类损失。
   * 特征对比损失。

**输出**：可靠的动作候选片段及置信度评分。

**阶段2：I\_Model**

**目标**：精细化调整动作边界并评估完整性

**核心组件**：

* Boundary\_Completeness\_Regressor：边界完整性回归器

**工作流程**：

1. **特征提取**：从候选片段中提取ROI特征。
2. **边界回归**：
   * 预测开始/结束位置的偏移量。
   * 计算片段完整性得分（IoU）。
3. **动态优化**：
   * refine\_proposals\_se()：边界调整算法。
4. **损失函数**：
   * 边界偏移平滑L1损失。
   * 完整性得分回归损失。

**输出**：精细化调整后的动作片段及完整性评分。

**训练流程（main.py）**

1. **阶段1训练**：
   * 初始化记忆库（memory.init()）。
   * 交替进行：
     + 特征增强与分类训练（S\_train()）。
     + 可靠性评估测试（S\_test()）。
   * 保存最优模型。
2. **阶段2训练**：
   * 加载阶段1生成的可靠候选片段。
   * 边界回归训练（I\_train()）。
   * 完整性评估测试（I\_test()）。

**关键创新点**

1. **双阶段弱监督**：
   * 阶段1：粗粒度动作定位。
   * 阶段2：细粒度边界优化。
2. **原型记忆库**（Reliable\_Memory）：
   * 动态更新类原型特征。
   * 解决弱监督中特征漂移问题。
3. **边界-完整性联合优化**：
   * 同步优化边界位置和完整性得分。
   * 公式：

loss\_total = weight\_loss\_reg \* reg\_loss + weight\_loss\_score \* iou\_loss

**文件结构说明**

|  |  |
| --- | --- |
| **文件** | **功能描述** |
| main.py | 主训练/测试流程控制器 |
| model\_S.py | 阶段1模型 |
| model\_I.py | 阶段2模型 |