北 京 邮 电 大 学

2023 届本科毕业设计（论文）中期进展情况检查表

学院	网络空间安全学院		专业	网络空间安全专业	
学生姓名	卢亭松	学号	2019212443	班级	2019211806
指导教师姓名	陆月明	所在单位	网络空间安全学院	职称	教授
设计（论文）题目	（中文）一种基于 FATE 框架的联邦学习方法设计与实现				
	（英文）Design and Implementation of a Federated-Learning Method Based on FATE Framework				

目前已完成任务	**主要内容：** 1. 学习机器学习、隐私保护相关的知识，对研究背景进行调研 2. 基于 FATE 搭建联邦学习集群实验环境 3. 基于 DSL 和 Pipeline 配置任务 4. 完成两种以上的联邦机器学习算法的传统实现与联邦环境实现，能够在 FATE 集群上进行训练 **详细设计：** **1. 学习机器学习、隐私保护相关的知识，对研究背景进行调研** 联邦学习的定义为：在进行机器学习的过程中，各参与方可借助其他方数据进行联合建模。各方无需共享数据资源，即数据不出本地的情况下，进行数据联合训练，建立共享的机器学习模型。 联邦学习系统需要保证：|联邦学习模型的效果-传统方法模型的效果|<有界正数。 联邦学习的价值机制：联邦学习技术基于“合作共赢”的价值机制，对于商业利益而言极具价值。在这样一个联邦机制下，各个参与者的身份和地位相同，而联邦系统帮助大家建立了“共同富裕”的策略，能够带动跨领域的企业级数据合作、催生基于联合建模的新业态和模式、降低技术提升成本和促进创新技术发展。 联邦学习在不同场景下的联邦方式：横向联邦学习、纵向联邦学习、联邦迁移学习 **2. 基于 FATE 搭建联邦学习集群实验环境** **a. 在 CentOS7 主机 x2、Docker 20.10.21 环境下完成 Fate 环境的部署** **b. 基于 DSL 配置初始任务验证 Fate 联邦学习流程（上传数据、构建模型、提交任务、部署模型、加载模型、测试模型）** • 进入 Host client 容器修改 examples/upload_host.json 并上传 host 数据 • 构建模型：修改 examples/job_conf.json 配置各个节点使用的算法组件及参数；修改 examples/job_dsl.json 配置算法组件间的输入输出流及接口，根据此拼接得到完整的工作模型

- 使用 fate-flow 提交任务并部署模型，并使用 POST 方式测试模型效果

c. **基于 DSL 配置初始任务验证 Fate 联邦学习流程（上传数据、构建模型、提交任务、部署模型、加载模型、测试模型）**

- 为 pipeline 配置关联的 FATE Flow Service 并配置相关 Python 环境
- 使用 Pipeline 上传数据：生成 Pipeline 实例并定义数据存储分区、表名和命名空间，之后使用 Pipeline 添加要上传的数据并执行数据上传
- 使用 Pipeline 完成 secureboost 训练：首先定义模型前工作流如下，Reader 组件加载数据； DataTransform 组件解析原始数据到数据实例中； Intersection 组件以计算联邦场景 PSI。然后定义 HeteroSecureBoost 组件创建模型结构，并定义 Evaluation 组件显示评估结果。最终添加组件到 pipeline 构建模型并编译
- 保存训练模型、加载并部署训练模型，执行预测任务验证模型效果

3. **完成两种以上的联邦机器学习算法的传统实现与联邦环境实现，能够在 FATE 集群上进行训练**

a. **基于 Pipeline 实现横向联邦 LSTM 完成 IMDB 文本情感分类，完成 LSTM 模型结构在联邦学习上的实现**

- 下载得到 IMDB 数据集，使用 tokenizer 将数据中的每个单词转化为整数值的唯一映射，词空间的最大个数不超过 10000，出现频率度低的词会被过滤；每个句子长度固定为 200，超过的部分将被截取，较短的句子将用 0 补齐
- 将数据集按联邦节点划分并输出到 csv 文件并基于 Pipeline 上传 IMDB 数据
- 基于 Pipeline 构建 LSTM 模型。构建 LSTM 模型词嵌入层设置 128 个神经元，LSTM 层包含 64 个神经元；最大迭代次数为 100，

	batch 长度为 32，设置 early_stop 机制。使用 Adam 进行梯度更新，学习率为 1e-5，损失函数为二元交叉熵 • 将定义好的 reader、data_transform、homo_nn、evaluation 结构通过 Pipeline 搭建组件结构并发布 • 保存刚才的算法模型组件 homo_nn_0，使用新的 Reader 读入测试数据集，搭建组件发布预测任务并验证效果 **b. 基于 Pipeline 实现横向联邦 CNN 完成中文文本主题分类** • 下载得到 THUCNews 数据集，读取词汇表并将每个值都转化为 unicode，通过 unicode 表将文件转换为 id 表示，同时使用 keras 提供的 pad_sequences 来将文本 pad 为固定长度 • 将数据集按联邦节点划分并输出到 csv 文件并基于 Pipeline 上传 IMDB 数据 • 基于 Pipeline 构建 CNN 模型。构建嵌入层 128 个神经元，CNN 层 32 个神经元，学习率为 1e-4，损失函数为二元交叉熵。其中 label_encoder 使用了 Fate 自带模块 • 将定义好的 reader、data_transform、homo_nn、evaluation 结构通过 Pipeline 搭建组件结构并发布 • 保存刚才的算法模型组件 homo_nn_0，使用新的 Reader 读入测试数据集，搭建组件发布预测任务并验证效果
	是否符合任务书要求进度　　是
尚需完成的任务	1. 基于 Pipeline 实现横向联邦 CNN 完成中文文本主题分类，完成 CNN 模型结构在联邦学习上的实现 2. 在 FATE 平台上完成预测与评估，对比分析算法在集中式环境与联邦环境的差异，从准确率，安全性，通信效率等进行分析 3. 阅读 FATE 框架源代码，学习工程设计思路并思考测试现存问题及其对应优化方向 4. 根据实验结果，完成论文的撰写
	是否可以按期完成设计（论文）　　是 ☑　　否 ☐

<table>
<tr><td rowspan="2">存在问题和解决办法</td><td>存在问题</td><td colspan="3">1. 对较复杂模型结构或较大规模数据进行处理时，容易内存溢出
2. 对当前 Fate 框架工程架构设计理解尚不明晰，导致在实验过程中遇到问题进行 Debug 缺少方向与日志
3. 联邦学习环境与集中式环境对比测试中需要对通信效率、计算效率进行对比评估，但尚未确定具体的评估标准与测试标准</td></tr>
<tr><td>拟采取的办法</td><td colspan="3">1. 通过租用线上 GPU 平台完成多机部署，搭建实验环境（如 AutoDL、Vast.ai 等平台）
2. 继续阅读 Fate 工程源代码，并总结其工程架构与设计思路
3. 参考传统或分布式机器学习框架优化计算性能方向的论文，学习其评估通信效率、计算效率的评估方法</td></tr>
<tr><td colspan="2">指导教师签字</td><td></td><td>日期</td><td>年　月　日</td></tr>
<tr><td colspan="2">检查小组评分及意见</td><td colspan="3">评分：　（总分：　）

组长签字：　年　月　日</td></tr>
</table>

注：可根据长度加页。