

人工智能与机器学习课程大作业选题

编程语言推荐 Python、C 等（鼓励基于华为 MindSpore 框架），以下设计均需要提供源代码（含说明）、设计报告（格式模板见附件）、演示视频等。

可分组，也可单人成组，每组限选以下选题；

每组组长需在最后的设计报告中明确每位组员的分工及工作量，按学校要求如果分组同学工作量相差过大，需有书面详细说明；

每组提交的设计报告最后需要有每位组员的签名（电子签名）

1、公交换乘系统设计及开发 (每组不超过 2 人)

自行获取（北京、上海、广州、深圳等）公交线路信息，包括地铁线路和公交车线路，获取方式不限。需要包含至少 50+ 公交（地铁线路），800+ 站点。

开发一个公交换乘系统，用户输入起点站和终点站，系统自动计算出换乘方案，即在哪一站上哪一路地铁或公交，在哪一站下接着换哪一路地铁或公交，直到终点。输入输出格式自行确定。**本选题主要侧重于人工智能相关算法实现，要求编程至少实现 2 个以上的已有算法（不是现有函数调用）。**

为了简化问题，只考虑用户起点和终点是公交或地铁站的情况，不用考虑用户如何到达公交地铁站点的问题，也不需要考虑两条线路的换乘站不是同一个站点需要步行完成换乘站上下车的情况，可以假设用户在换乘站下车后零距离登上下一路公交或地铁。也不需要考虑实际行车距离，可以假设每一站的长度都一致，经过的站点越多，行车距离越远。

地铁和公交线路视为等同优先级的交通方式，在算法设计中不需要对其差异进行特殊考虑。公交站名和地铁站名相同时可认为站点在同一个地方，用户无需步行即可原地完成换乘。

计算线路方案时考虑两个优化的目标：

1) 用户追求最短距离优先，要求乘车方案经过站点数最少，不考虑换乘次数，但在有多个经过站点数量相等的最优方案时，换乘次数最少的方案胜出，若仍有平行方案，则输出这些方案供用户选择。

2) 用户追求换乘次数最少，不考虑经过站点数量的多少，但在有多个换乘次数相等的最优方案时，经过站点最少的方案胜出，若仍有平行方案，则输出这些方案供用户选择。

开发环境和编程语言不限（如使用华为 MindSpore 框架，有额外加分），**若使用他人或者开源代码需标明出处。**

程序运行结果有对比（不同算法间，以及与百度地图或高德地图的输出结果进行对比，并对差异的原因进行说明）。对比指标可以包括规划换乘方案的优劣，也可以包括算法执行效率、时间空间复杂度等。

以上基本功能完成后，如有创新性改进，有额外加分。

2、联邦学习仿真框架的实现与改进（基于树莓派、手机等边缘设备的 FedML 平台搭建与实现）(不超过 2 人)

Federated Learning 联邦学习是机器学习领域中快速发展的研究领域。尽管已经进行了大量的研究工作，但是现有的软件框架不能充分支持多样化的算法开发（例如，多样化的拓扑和灵活的消息交换），并且实验中不一致的数据集和模型使用使公平的比较变得困难。

而 FedML 就能非常好的解决上述问题，它是一个开放的研究库和基准，可促进新的联邦学习算法的开发和公平的性能比较。FedML 支持三种计算范例：分布式训练，移动设备训练

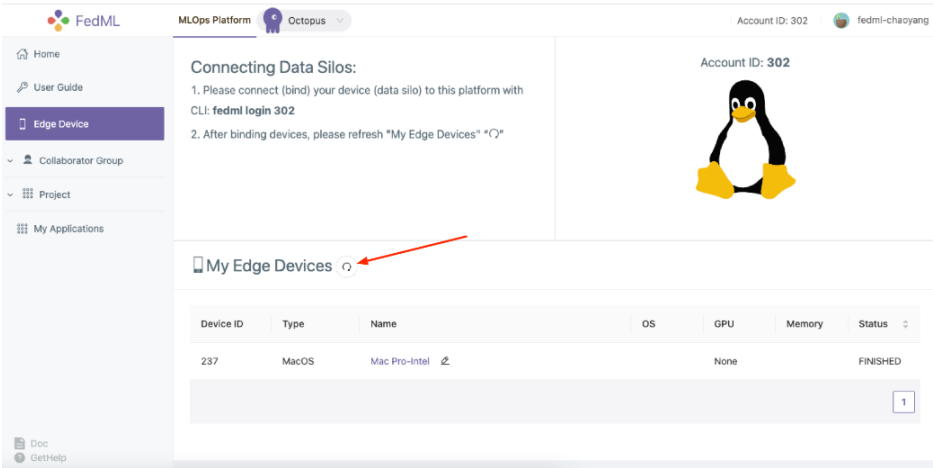
和独立仿真，以便在不同的系统环境中进行实验。FedML 还通过灵活且通用的 API 设计和参考基准实现促进了各种算法研究。

要求至少在 2 台及以上树莓派等边缘设备上搭建 FedML 平台，树莓派的数量可以根据小组人数确定，FedML 安装教程可参考 FedML 官方文档，安装完成后需要执行以下代码并将程序运行结果截图：

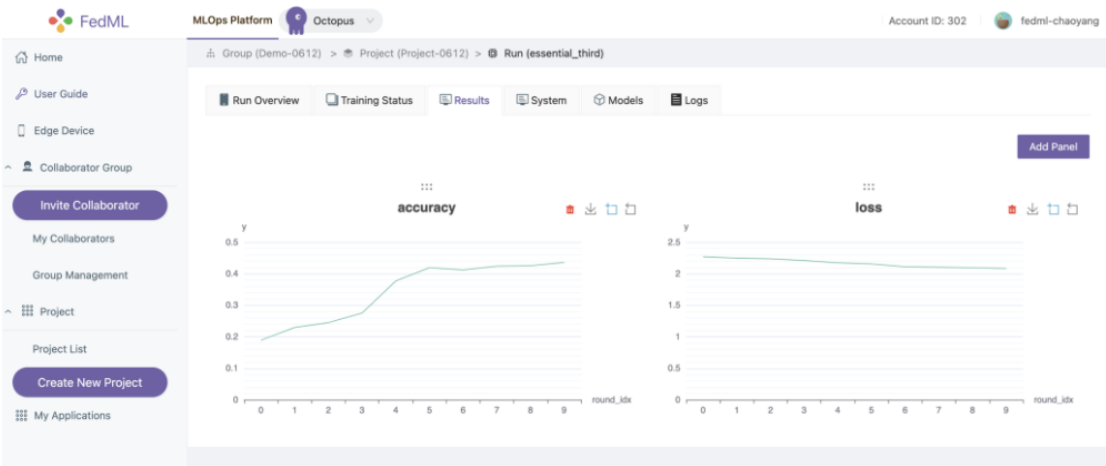
```
import fedml

if __name__ == "__main__":
    fedml.run_simulation()
```

安装完成后使用 FedML 的 MLOps 平台完成一个分布式联邦学习训练任务，登陆网址 <https://open.fedml.ai> 进行注册获得一个 AccountID，注册完成后创建一个组并邀请组员进入组中，截图显示每个组员。其中各树莓派视为客户端，服务端可以使用 FedML 提供的免费的公有云聚合服务器，也可以使用自己的笔记本作为服务端。其中训练任务的模型和数据集可自由选择，但不得使用官方提供的模型，必须要自己创建或改进原有模型，数据集可使用官方提供的也可使用自己的数据集。在树莓派中使用命令 `fedml login` 登陆 MLOps 平台，在 Edge device 界面截图显示自己的设备。（如下图所示）



在任务训练界面需要截图展示设备状态、训练结果（accuracy、loss）、系统性能等指标，如下图所示：



以上基本功能完成后，根据训练任务的复杂度及模型的性能进行一定程度的加分。

FedML 源码 Github 地址: <https://github.com/FedML-AI/FedML>

FedML 官网主页: <https://fedml.ai/>

FedML 官网教程: <https://doc.fedml.ai/>

FedML 论文: 《FedML: A Research Library and Benchmark for Federated Machine Learning》

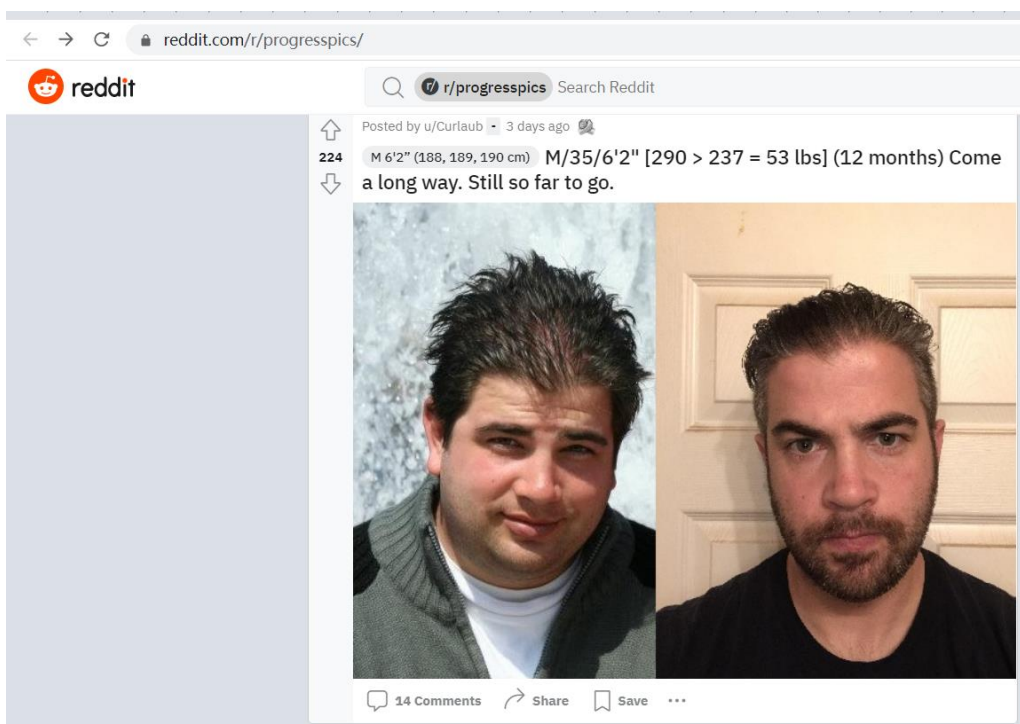
下载地址: <https://arxiv.org/abs/2007.13518>

3、基于深度学习的人脸-BMI 预测系统与实现（不超过 3 人）

BMI (Body Mass Index) = 体重(kg) / 身高(m²), 是评价肥胖的一个世界公认的评定标准, 一个人的 BMI 值越高就代表这个人的肥胖程度越高。随着人的体重变化, 面部形状也会发生变化。据研究表明, 面部肥胖与 BMI 是呈正相关的: 随着 BMI 的增大, 面部肥胖程度也开始增大, 也与血管疾病和糖尿病的风险增大呈正相关, 因此可以基于人脸图像进行 BMI 的预测。本项目拟采用深度学习实现基于人脸图像进行 BMI 的预测。

项目内容包含:

- (1) 数据搜集: 通过社交网络 <https://www.reddit.com/r/progresspics/> 爬取数据构建数据集, 该网站的用户上传前后变化的两张图及身高体重, 学生需要自己写爬虫下载人脸和身高体重等数据, 下载数据后需用人脸检测模型提取人脸。该步骤也可采用公开数据集, 但自己爬取数据会加分;



- (2) 模型搭建和训练: 可使用任何深度学习框架, 如 Pytorch、Tensorflow 等, 需要自己构建模型, 读取数据, 进行训练, 并保存模型。训练方式可以完全从头训练, 也可以导入在 ImageNet 上训练好的模型, 如 ResNet 等, 也可以导入人脸识别模型, 如 FaceNet 等, 视学生进行的实验丰富程度酌情加分。

```

LR optimization Passes are enabled (registered 2)
Epoch 1/8
52/52 [=====] - 124s 2s/step - loss: 4.4253
Epoch 2/8
52/52 [=====] - 118s 2s/step - loss: 3.0853
Epoch 3/8
52/52 [=====] - 118s 2s/step - loss: 2.6978
Epoch 4/8
52/52 [=====] - 117s 2s/step - loss: 2.5270
Epoch 5/8
52/52 [=====] - 137s 2s/step - loss: 2.3360
Epoch 6/8
52/52 [=====] - 117s 2s/step - loss: 2.2641
Epoch 7/8
52/52 [=====] - 125s 2s/step - loss: 2.3285
Epoch 8/8
52/52 [=====] - 125s 2s/step - loss: 2.3514
Mean squared error = 2.970698595046997

```

- (3) 模型部署和系统搭建：建议采用 Flask 构建界面和后端，在后端建议采用 ZMQ 通信，后端读取第(2)训练的模型，等待前端图像的导入，前端用户上传包含人脸的图像，后端需对人脸图像进行人脸检测，建议采用第一步(1)的人脸检测工具，得到人脸后根据训练的模型进行预测，并把结果返回前端。



以上基本功能完成后，根据训练任务的复杂度及模型的性能进行一定程度的加分。

4、体现人工智能的自选项目（比照上述难度和工作量，需教师认可）

- （1）自选一个利用人工智能技术求解实际问题的小项目；
- （2）要有明确的人工智能算法实现；
- （3）要有良好的人机界面。