

**《云计算与虚拟化》**

**论文读后感**

**专业班级：** 计算机科学与技术2008班

**学 号：** U202015533

**姓 名：** 徐瑞达

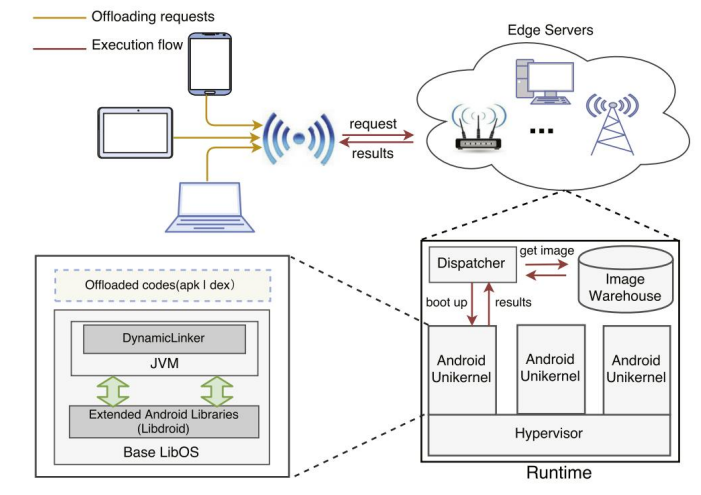
**教 师：** 吴松

**文献题目：** Android Unikernel: Gearing mobile code offloading towards edge computing

听了吴老师的云计算与虚拟化这门课，我认识到云计算服务的灵活和高效。同时，我更加希望了解云计算在日常生活中的应用场景，因此选择了Android Unikernel: Gearing mobile code offloading towards edge computing这篇论文来研究如何通过计算卸载提高终端设备计算能力并延长其电池寿命。

计算卸载这一技术最初在移动云计算MCC中提出——由于移动云计算具有强大的计算能力，则可以将一些计算密集型任务卸载到集中式云服务器上运行并返回计算结果。然而这种实现方式的性能受限于网络延迟和服务器启动延迟，从而会导致高延迟响应。因此提出了边缘计算的概念，即将移动终端的计算任务卸载到边缘云环境中，降低网络延迟的同时也缓解了核心网络的压力。然而，MCC架构中使用的传统虚拟机/容器的启动延迟仍旧较大，因此需要寻找一个轻量级的运行环境以支持计算卸载。这篇论文中，提出了一种解决方案——在Unikernel的基础上，通过动态链接构建Android Unikernel使即时计算卸载服务成为可能。

该解决方案的主要架构如下图所示：



首先分析不同类型的虚拟化技术的优点和缺点。传统的Android-X86系统虚拟机对于资源受限的边缘服务器太过重量级，同时其启动时间过长，无法提供即时服务。为了保证服务质量提前启动服务器又会造成资源浪费。为此，提出了轻量级的虚拟化技术如容器，大大减小了云端开销但其隔离性和安全性较低。Unikernel则是一种小而快的技术，通过将一个应用连同其必要的系统库组织成独立镜像消除虚拟机和容器的缺点，可以尝试将其用于计算卸载中。

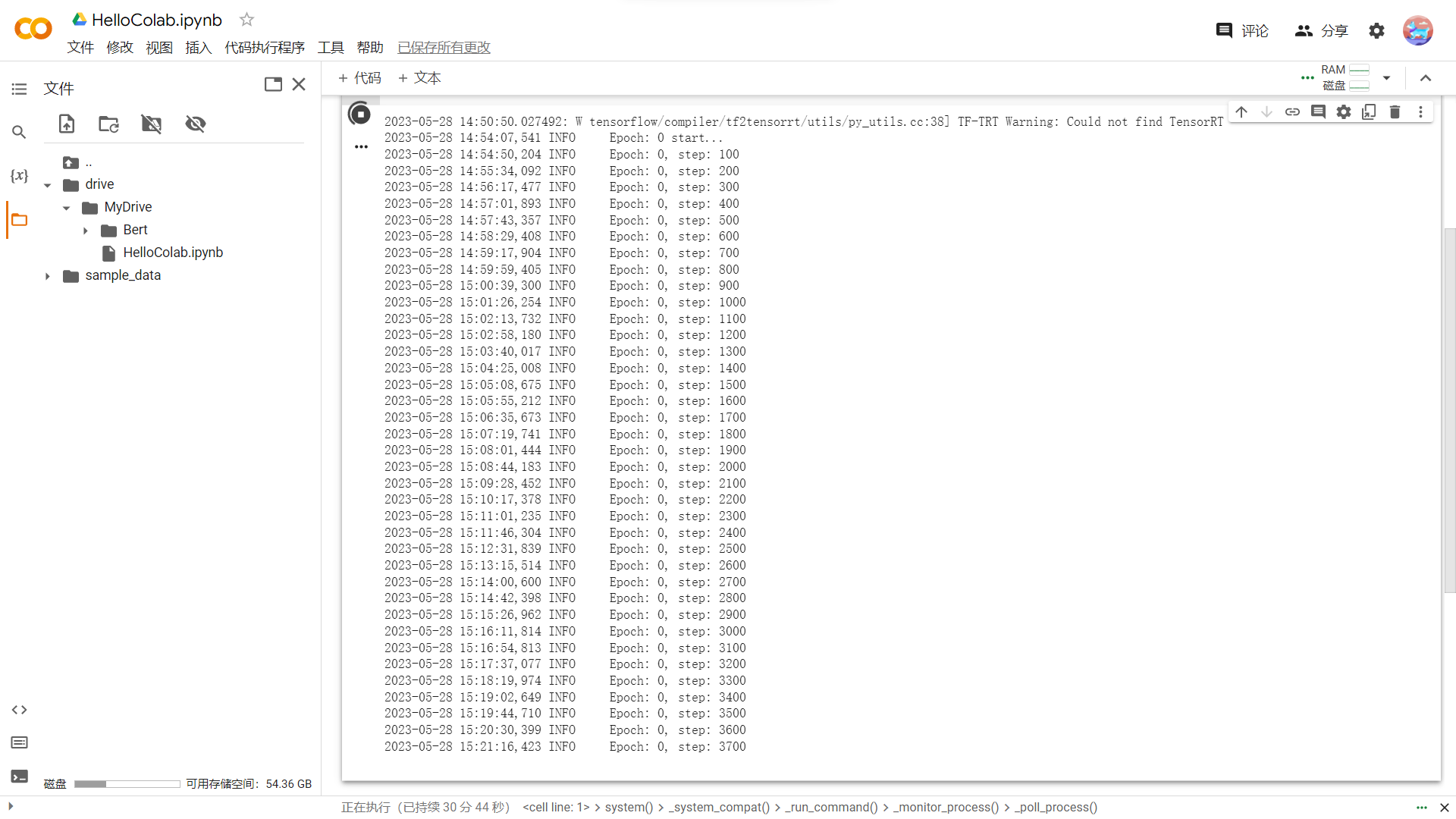
然后分析使用Unikernel进行计算卸载的难点。由于Unikernel是专用于某个应用的不可变镜像，因此需要为每个应用构建镜像；由于Unikernel构建过程较慢，因此需要寻求方法在卸载请求到达后迅速构建镜像；由于Unikernel专用于Linux系统，因此需要让Unikernel运行Android代码。

对于第一个挑战，通过构建具有一系列公共库和Application Loader的Rich-Unikernel实现动态的Unikernel，同时使用Image WareHouse存储预构建的Unikernel保证即时服务质量；对于第二个挑战，通过DynamicLinker动态链接Android代码；对于第三个挑战，通过将必要的Android系统库组织成独立的Libdroid模块移植到Android Unikernel使其能够运行Android代码。

最后部署安卓客户端与模拟的云计算服务器，分别比较在虚拟机、容器、Android Unikernel下的启动延迟、系统开销、设备能耗和计算性能。

在阅读完论文后，我尝试部署了Libdroid代码。首先在WSL中安装osv环境，克隆项目并编译Android Unikernel，服务端部署成功。然后编译Android客户端，但由于客户端版本较低，不适配手机系统，最终无果。

为了体验云计算的强大之处，我了解了国内外云服务提供商提供的服务。其中，Google Colab通过按需分配的云计算资源提供方式和虚拟化技术为开发人员提供了强大的GPU算力，且国内百度的飞桨平台也提供了自带Paddle环境的深度学习计算平台。通过使用Google Colab平台提供的云GPU资源并配合使用Google Drive上传项目文件，能够大大提升使用预训练模型Bert进行中文分词等深度学习任务的训练速度，如下图所示。



如图，在使用15GB的GPU算力后，运行3700次梯度下降运算（大规模矩阵运算）只需要半小时即可计算完成，而在本地只使用CPU运行时，则需要花费数十个小时之久。

通过阅读论文和上手实践，我认识到了科研开发的一种途径——通过不断尝试优化既有架构来提高性能，也体会到了云计算与虚拟化的强大之处。

在课程的过程中，老师主要侧重于讲述关于云计算与虚拟化的基础知识，如从基础设施即服务Laas到平台即服务Paas再到软件即服务Saas的发展过程等内容，在理论层面上极大地拓宽了我的视野。当然，也希望老师能够提供有关云计算的实践途径，如在腾讯云、阿里云或华为云等国内云服务商上部署个人云服务器、容器镜像等服务。