RAFAEL MARTINS SANTOS COSTA

BSc in Computer Science and Engineering

TOWARDS DECENTRALISED SPLIT LEARNING

SELF-MANAGING ARCHITECTURE FOR COLLABORATIVE DECENTRALIZED MACHINE LEARNING

TOWARDS DECENTRALISED SPLIT LEARNING

SELF-MANAGING ARCHITECTURE FOR COLLABORATIVE DECENTRALIZED MACHINE LEARNING

RAFAEL MARTINS SANTOS COSTA

BSc in Computer Science and Engineering

Adviser: João Leitão
Professor, NOVA University Lisbon

Co-adviser: Dimitra Tsigkari
Research Scientist, Telefónica Research (Spain)

Abstract

In recent years, the rise of data-driven applications has driven the development of decentralized learning frameworks such as Federated Learning (FL) and Split Learning (SL), designed to improve data privacy and distribute computational load across multiple devices. However, current FL systems still rely heavily on centralized aggregators, which imposes privacy and efficiency limitations due to the centralized orchestration..

This thesis proposes a fully decentralized platform that combines FL and SL methodologies to overcome these challenges. Our approach designs a self-managed system for collaborative decentralized machine learning, minimizing human intervention through dynamic delegation protocols and adaptive role assignment mechanisms. Participants can collaboratively train machine learning models in a secure environment, with devices autonomously handling changes in the network, such as device failures and connectivity changes.

Our research contributes to the field of decentralized learning by providing a robust alternative to traditional centralized systems, enabling collaboration and adaptability.

Keywords: decentralized platform, FL, SL, collaboration, adaptability

Resumo

Nos últimos anos, o aumento de aplicações baseadas em dados impulsionou o desenvolvimento de estruturas de aprendizagem descentralizadas, como a Aprendizagem Federada (FL) e a Aprendizagem Dividida (SL), concebidas para melhorar a privacidade dos dados e distribuir a carga computacional entre vários dispositivos . No entanto, os atuais sistemas de FL ainda dependem fortemente de agregadores centralizados, o que impõe limitações de privacidade e eficiência devido à orquestração centralizada.

Esta tese propõe uma plataforma totalmente descentralizada que combina as metodologias FL e SL para ultrapassar estes desafios. A nossa abordagem projeta um sistema autogerido para aprendizagem automática descentralizada e colaborativa, minimizando a intervenção humana através de protocolos de delegação dinâmicos e mecanismos adaptativos de atribuição de funções. Os participantes podem treinar modelos de aprendizagem automática de forma colaborativa num ambiente seguro, com os dispositivos a lidarem autonomamente com alterações na rede, como falhas de dispositivos e alterações de conectividade.

A nossa investigação contribui para o campo da aprendizagem descentralizada, fornecendo uma alternativa robusta aos sistemas centralizados tradicionais, permitindo a colaboração e a adaptabilidade.

Palavras-chave: plataforma totalmente descentralizada, FL, SL, colaboração, adaptabilidade

CONTENTS

LIST OF FIGURES

Acronyms

1

INTRODUCTION

In recent years, the exponential growth of data generated through distributed devices and platforms has driven the development of collaborative learning methods such as Federated Learning (FL). Federated learning allows a network of participants to train a machine learning model without centralizing data, addressing critical issues related to data privacy and security. However, FL still faces inherent limitations, particularly in its dependence on a central aggregator and the computational resources required on individual devices to train complex models. Furthermore, reliance on centralized platforms raises significant privacy concerns as they can introduce vulnerabilities or potential access points to sensitive data. Alternatively, Split Learning (SL) has emerged as a promising approach within distributed learning paradigms, allowing computational load to be shared among participants, which can help alleviate resource constraints and increase data privacy by increasing sensitive data that remains on the source device.

The objective is to design, implement and evaluate a fully self-managed and decentralized platform capable of leveraging SL techniques, allowing participants to independently configure and reconfigure as needed, even in the presence of dynamic network conditions.

This decentralized platform seeks to minimize human intervention and ensure robustness by allowing participants to help each other in the training process, thus addressing privacy concerns that arise from centralized architectures.

This research will first address the current state of the art in decentralized learning systems and evaluate existing methods, followed by the design of a decentralized SL architecture. Through this approach, we aim to improve the efficiency and privacy of federated systems, ultimately contributing to a more robust infrastructure for distributed learning.

1.1 Objective

The objective of this thesis is to develop a fully decentralized federated learning platform that integrates key principles of Split Learning (SL) to enhance privacy, efficiency, and robustness. By moving away from reliance on a central aggregator, this research aims to

build a secure and autonomous framework where participants can collaborate without compromising data privacy.

To achieve this, the study will focus on the following key areas:

1. **Decentralized Federated Learning:**

   Design a federated learning system that operates in a decentralized manner, reducing reliance on central servers and enabling greater autonomy for participants.

2. **Privacy:**

   Implement privacy-preserving mechanisms that ensure sensitive data remains secure on the source devices, minimizing data exposure risks within the learning process.

3. **Split Learning:**

   Incorporate split learning techniques to distribute computational loads, allowing resource-constrained devices to participate effectively in the model training without overextending their capabilities.

4. **Delegation Protocols:**

   Develop delegation protocols that allow to assign an idle node with enough processing capabilities available as a helper to train a model.

5. **Helper Membership Management:**

   Establish mechanisms for helper membership management, enabling participants to support one another in the training process and foster a robust, collaborative learning environment.

1.2 Expected contributions

.....

1.3 Research Context

In the context of the European TaRDIS Project, this thesis aims to explore a new decentralized approach to implementing Divided Learning within a Federated Learning framework.

... Movistar musica - Telefónica ....

.....

1.4 Thesis Structure (Roadmap)

The rest of the document is organized as follows:

1. Chapter 2 (Related work)

   This chapter provides a review of the foundational concepts of Federated Learning (FL) and Split Learning (SL), examining their architectures, strengths, and limitations. It also explores existing applications, delegation protocols for task distribution, and strategies for managing helper membership.

2. Chapter 3

   Presents a . . . .

2

RELATED WORK

In this chapter we further detail the issues related to the Centralized Federated Learning Applications and explain why we should use Decentralized Federated Learning Applications as well as how are we expecting to deal with some issues related to delegating helpers to improve systems performance and avoid adverse effects on the functionality of the devices (the network nodes), but first some fundamental subjects will be introduced for a better general understanding of our proposal.

2.1 Decentralized Systems

Decentralized systems are fundamental in distributed computing, enabling tasks to be executed without a central authority.This architectural approach enhances scalability, fault tolerance, and resilience by distributing responsibilities across multiple nodes, allowing them to function autonomously while maintaining coordination. Such systems are particularly valuable in environments that require adaptive and robust communication between a multitude of participants, including Peer-To-Peer (P2P) networks, blockchain technologies, and decentralized applications.

In decentralized systems, nodes can dynamically join, leave, or fail without significantly disrupting overall functionality.This is made possible by the use of adaptive protocols that ensure that the system remains operational. Data sharing, connectivity maintenance, and task distribution are facilitated by the deployment of algorithms and protocols. For example, membership protocols guarantee that nodes are aware of each other's presence, whereas peer-to-peer communication models facilitate direct interaction between nodes. Membership protocols serve to ensure that nodes are aware of each other's presence, while P2P communication models facilitate direct interaction between nodes.

Decentralized systems are a foundational element of modern technologies, including blockchain and decentralized P2P file-sharing platforms. These systems demonstrate the potential of decentralization to address scalability and reliability challenges in diverse domains.

2.1.1 Membership Protocols

Membership protocols are vital for keeping decentralized swarm systems organized. These systems rely on many individual nodes that interact to accomplish tasks, and each node needs to know which other nodes are active at any moment.

1. **Epidemic and Scalable Global Membership Service:**

   The Epidemic Membership Service works much like spreading a rumor: each node shares information about its presence with a few nearby nodes. These nodes then pass along the update, which eventually reaches the whole network. This approach enables each node to keep a "partial view" of the network, meaning it stays updated on some nodes but not all. This setup keeps the system scalable and ready to handle changes in swarm size. Since updates move organically throughout the network, nodes can join or leave without burdening any single part of the system.

2. **Decentralized Membership Protocols:**

   In larger swarms, keeping track of every single node isn't practical. Decentralized Membership Protocols simplify this by letting each node manage enough information to stay connected without needing to know all nodes. For example, the HyParView protocol organizes nodes into two lists: an active view, which contains frequently updated links to a few critical nodes, and a passive view, a backup list of other nodes. The active view keeps each node actively connected, while the passive view acts as a fallback in case of node failure. This two-layered approach makes HyParView resilient and allows quick recovery if nodes unexpectedly drop out. The protocol is often used within frameworks like Babel, which supports secure and automatic connections.

2.1.2 P2P communication

Peer-to-peer (P2P) communication is a decentralized network model in which each peer or node has equal capabilities and responsibilities. P2P enables direct communication between nodes, without needing a central authority to mediate the interaction. Nodes can connect and disconnect dynamically leading to an architecture that adapts as nodes join or leave the network. These types of networks can scale up or down based on the number of peers. Increasing the number of nodes improves the network's capacity and resilience and decreasing them has limited impact, assuming enough nodes remain active to support data flow. Each node can offer resources, such as processing power or storage capacity, contributing to the overall capacity of the network, this makes P2P an efficient approach for apps requiring vast amounts of shared data.
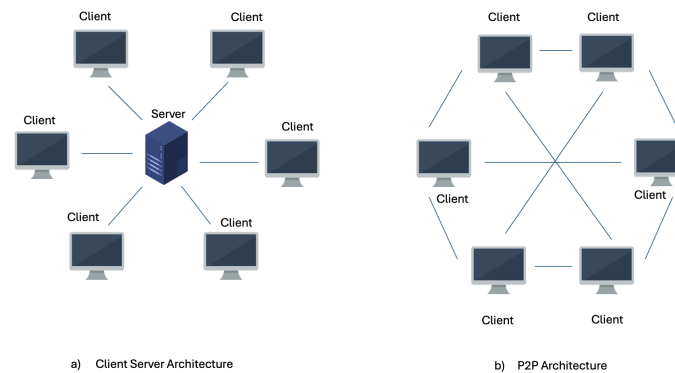
Figure 2.1: Illustration of client-server and P2P architectures

2.1.2.1 Broadcast

The term broadcast refers to sending messages or data to all peers within a network, since P2P networks lack a centralized server, broadcasting relies on each node to propagate the message to others. This method allows information to spread across the network efficiently, but it also presents challenges related to network congestion, duplication, and ensuring all peers receive the broadcasted message.

Broadcasting can be implemented using various techniques:

1. **Flooding:**

   In flooding, a peer sends the message to all its directly connected peers, who then forward it to their connected peers, and so on, until the message reaches every peer in the network. This method can lead to message duplication and network congestion, as each peer may receive multiple copies of the same message.

2. **Gossip Protocols:**

   Each node randomly selects a few nodes to send the message, and those nodes, select randomly another subset of their connections, and so on. This method ensures that most or all nodes receive the message without overloading the network.

3. **Multicasting:**

   In structured P2P networks, multicasting can be more efficient than broadcasting. Instead of randomly sending messages to every node, it sends only to a specific group/subset of nodes who are most relevant to receive it. This is done by organizing nodes into groups or clusters and sending messages within those or between them.

2.1.3 Decentralized development Frameworks

The design and implementation of distributed systems are inherently complex due to inherent challenges such as fault tolerance, concurrency, and efficient communication. In order to address these difficulties, researchers and practitioners have developed a number

of frameworks which abstract low-level details, enabling developers to focus on the core logic of distributed protocols. These frameworks differ in their design philosophies, varying from educational tools for simulating algorithms to robust platforms for real-world deployment.

This section presents a review of several prominent decentralized development frameworks, namely Appia, Cactus, Yggdrasil, ViSiDia, and Babel, identifying features, strengths, and limitations of these frameworks in supporting the development of dependable distributed systems.

1. **Appia:**

    A Java-based toolkit leveraging inheritance for creating custom protocols. It introduces the concept of channels and sessions as abstractions, which are used to bind protocols into services. However, this approach requires developers to manually stack protocols, which restricts the flexibility of the interaction. Appia uses a single execution thread for all protocols, which can cause significant performance bottlenecks.

2. **Cactus:**

    A C++ framework designed to achieve optimal performance and expressiveness. It uses meta-protocols composed of smaller protocols that execute asynchronously, offering modularity. However, it does not manage concurrency, leaving developers responsible for addressing these challenges.

3. **Yggdrasil:**

    A lightweight and efficient C framework, designed for use in both wireless ad hoc and traditional wired IP networks. The framework is designed to facilitate the processing of four distinct types of events: network messages, timed events, direct interactions within the protocol, and indirect interactions. It supports concurrent protocol execution while abstracting concurrency complexities. However, Yggdrasil only supports a single active network abstraction per instance (e.g., wireless or wired), and its reliance on C necessitates rigorous technical discipline to prevent issues such as invalid memory access.

4. **ViSiDia:**

    A Java-based framework designed for understanding and visualizing distributed algorithms. It facilitates the design and simulation of these algorithms, making it an educational tool. The framework's primary objective is to facilitate the comprehension of algorithms, rather than to assess their performance in real-world scenarios or to analyze their interactions with protocols.

5. **Babel:**

Babel is a framework that promotes event driven programming and it helps in the development of distributed protocols within a process that can execute any number of different protocols that communicate with each other, inside or outside the same process. The system is designed with a generic architecture that allows for the implementation of multiple distributed protocols and provides networking components that can capture different capabilities (e.g., P2P, Client/Server, ...). It facilitates the development process by allowing the developer to concentrate on the fundamental aspects of the protocol, without having to address the inherent complications associated with low-level details.

2.1.3.1 Discussion

We opted to use Babel, since... ...
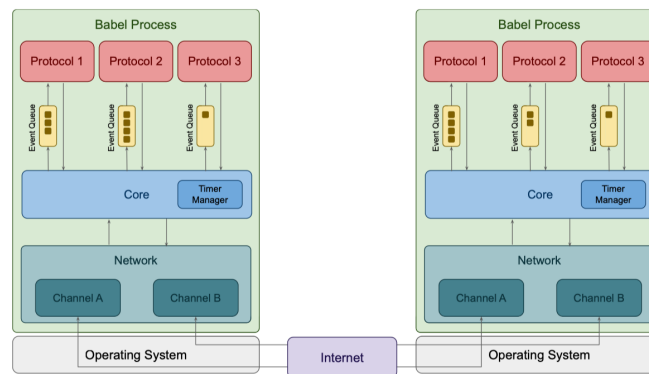
Figure 2.2: Architecture of Babel

Image source: Babel: A Framework for Developing Performant and Dependable Distributed Protocols

2.1.3.2 Babel swarm

....

2.2 Federated Learning (FL)

In recent years, Federated Learning (FL) has emerged as a relevant approach for training collaborative models without the need to share sensitive data. Since its emergence, Federated Learning (FL) (Centralized Federated Learning (CFL)) has been the predominant methodology in the extant literature, wherein a central entity is responsible for the creation of a global model. However, a centralized approach has the disadvantage of increasing latency due to bottlenecks, heightening vulnerability to system failures, and giving rise to concerns regarding the trustworthiness of the entity responsible for creating the global model. Decentralized Federated Learning (DFL) emerged to address these concerns

by promoting decentralized model aggregation and minimizing reliance on centralized architectures.

Figure 2.3: Illustration of local learning, centralized learning, CFL, and DFL

Image source: Decentralized Federated Learning: A Survey and Perspective

2.2.1 Centralized approach

In CFL, a central server is tasked with aggregating updates from local models trained independently on each participant's data. By exchanging model parameters only and not raw data, this configuration protects data privacy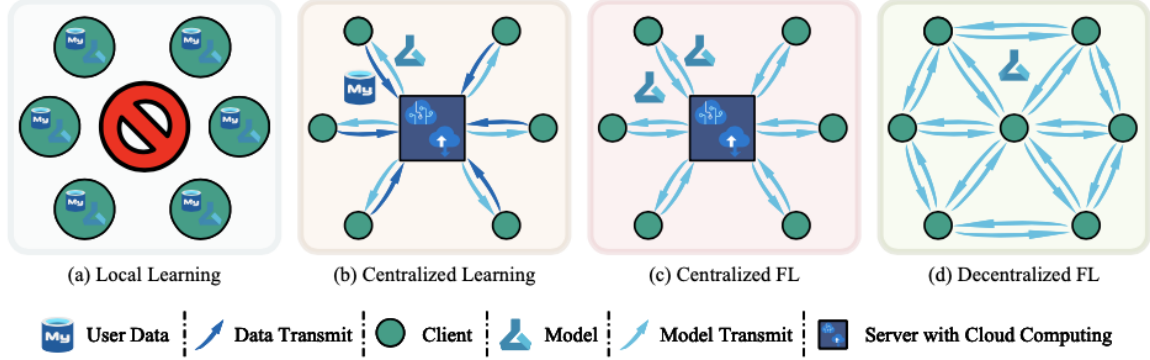. However, the dependency on a single aggregation server introduces several limitations. Firstly, as the number of participating devices rises, the central server may become a bottleneck, resulting in increased latency and reduced scalability. Additionally, this architecture introduces a potential risk of single-point failure, where the central server's failure can compromise the integrity of the entire training process. Security and trust are further concerns, as participants must trust the central entity with aggregating and distributing model updates without compromising data privacy.

Due to its simple design and well-established protocols for safe and effective aggregation, CFL is still widely used despite all of these disadvantages. In order to minimize the possibility of data leakage, optimization efforts in CFL frequently concentrate on lowering communication costs between participants and the server, enhancing the aggregation algorithms, and guaranteeing strong data privacy techniques, such as differential privacy and secure aggregation methods.

2.2.2 Decentralized approach

Decentralized Federated Learning (DFL) reduces dependency on a central server, which helps to overcome some of the main limitations of CFL. Since model parameters are shared and aggregated in a distributed manner rather than via a single central aggregator, DFL distributes computational responsibility more widely throughout the network.

However, even without a central server, DFL still maintains a mechanism for updating a global model through decentralized aggregation strategies, often involving clusters or peer-to-peer structures where nodes collaborate to aggregate model parameters locally before updating the shared model.

This decentralized approach minimizes bottlenecks and reduces the risk of single points of failure, making DFL inherently more fault tolerant and scalable. Additionally, it distributes trust across the entire network rather than concentrating it in a single entity, which can alleviate some concerns about trustworthiness in centralized environments. However, because node interactions are asynchronous, DFL makes it more difficult to maintain consistency in model updates, which can result in stale updates and slower convergence. It also increases communication overhead, as nodes must frequently exchange model parameters with their peers to keep the overall model aligned.

Given that there is no central authority, it can be difficult to detect malicious participants during DFL, making security and privacy issues difficult to resolve. For DFL systems to reduce the risk of malicious attacks or data exposure, secure aggregation, anomaly detection, and model update verification require strong, decentralized protocols. These mechanisms are essential to maintain the integrity of the model and the privacy of participants in the decentralized framework.
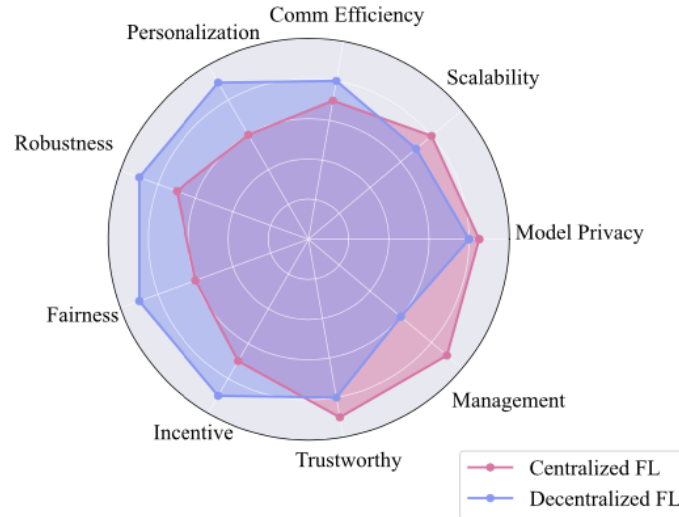

Figure 2.4: CFL versus DFL

Image source: Decentralized Federated Learning: A Survey and Perspective

2.2.3 Discussion

Considering these two approaches, we will focus on Decentralized Federated Learning (DFL) to ensure that personal data stays on local devices and is not shared across the network. With DFL, only model updates such as parameters and gradients are shared, not

the actual data, which helps protect privacy more effectively than a centralized approach. By choosing DFL, our goal is to create a learning network that respects user privacy while remaining resilient and efficient, even as it grows.

2.3 Decentralized Federated Learning (DFL)

DFL represents an extension of FL that eliminates the necessity for a central coordinating server. In contrast, participants communicate directly with one another in a P2P network to train a global machine learning model.

2.3.1 DFL Architecture

DFL is built on a P2P architecture where clients collaborate directly without a central server. Each client trains a model on its own data set and disseminates updates to other participants. This design enables the learning process while simultaneously preserving data privacy and avoiding the bottlenecks that are typical of centralized systems. Clients are autonomous entities that determine their own training schedules and communications based on their available resources and specific requirements.

Communication between clients is essential in DFL. Clients exchange model updates using protocols like one-to-one (direct communication), gossip (randomized sharing), or broadcast (one-to-many). These exchanges follow a network structure or topology, which defines how clients are connected. Common topologies include linear (sequential sharing), ring (cyclical sharing), and mesh (all-to-all communication). More complex hybrid structures combine these approaches to balance efficiency and flexibility.

Two principal approaches inform the manner in which models evolve in DFL. The Continual Paradigm involves the transfer of a model between clients, with each client updating the model using their own data. This approach is straightforward and resource-efficient, with a primary focus on personalization. In contrast, the Aggregate Paradigm involves the collection and combination of updates from multiple clients prior to local training. This method results in the generation of a more generalized model, although it demands higher levels of computational resources.

The workflow of DFL is fundamentally iterative. Clients begin with an initial model, either shared or independently created and follow a loop of training, sharing, and refining. Communication follows the chosen protocol and topology, ensuring that updates propagate effectively across the network. This cycle continues until the models reach convergence, where further updates no longer significantly improve performance. The decentralized structure makes DFL adaptable and resilient, capable of handling dynamic environments where client availability or data distribution might change.

Dynamic features, like adaptive topologies, allow DFL networks to evolve based on external factors, such as client participation or data shifts. Security measures, such as blockchain and differential privacy, are incorporated to ensure the trustworthiness of the

system and to safeguard against potential threats, including data leakage and malicious attacks.

DFL provides a robust, scalable, and privacy-preserving framework for distributed learning. Its architecture makes it particularly suited for applications requiring secure and efficient collaboration, such as healthcare, Internet of Things (IoT), and social networks.

2.3.2 Challenges in federated learning

Both CFL and DFL approaches, while innovative in preserving data privacy during collaborative learning, face several critical challenges:

1. **Device and data heterogeneity:**

   Participating devices in FL differ significantly in terms of available memory, network connectivity, and processing power. Additionally, data from different devices is frequently non-IID (not independently and identically distributed), meaning that each device's data may exhibit distinct patterns. As local data from each node can differ from the population as a whole, this lack of uniformity makes it more difficult to create an accurate global model. In DFL, the added challenge of device diversity requires adaptive methods for synchronising and aggregating model updates.

2. **Security and privacy:**

   Although FL naturally encourages data privacy by storing data locally, it is still susceptible to a number of attacks because client's shared parameters and gradients may unintentionally reveal sensitive information. Techniques like model inversion (machine learning security threat that involves using the output of a model to infer some of its parameters or architecture) or membership inference attacks (allows an adversary to query a trained machine learning model to predict whether or not a particular example was contained in the model's training dataset) can exploit these shared updates to reconstruct aspects of the original data or identify if specific data points were involved in training.

   These vulnerabilities demonstrate the necessity of strong privacy-preserving techniques, like secure aggregation or differential privacy, to reduce data leakage and preserve model performance.

3. **Scalability and fault tolerance:**

   While DFL enhances fault tolerance by eliminating single points of failure, scaling DFL networks remains challenging. As the network expands, the demands for communication and synchronisation among nodes grow, which can strain system performance. To manage large numbers of participants efficiently, strategies like effective node clustering, load balancing, and advanced fault tolerance techniques become essential.

These methods help distribute workloads more equally across the network and prevent communication bottlenecks, ensuring that model accuracy and system efficiency are maintained as the network scales

2.3.3 Models

....

2.3.4 Real world Applications

DFL framework development depends on various key factors, such as relevant application scenarios, sources of information acquisition, information processing units and perceptual prediction modules, among others. And so DFL has been adopted in several areas, such as, automotive industry, healthcare, IoT industry, social networks, etc.

1. **Connected and Automated Vehicles:**

   Connected and Automated Vehicles (CAVs) serve as a robust hardware infrastructure for DFL, where each node uses integrated batteries, various sensors, computing units, storage devices and much more. Vehicle-to-vehicle (V2V) communication is a framework that already supports several applications in networks of independent and connected vehicles, creating a basis for the application of Distributed Federated Learning (DFL) in Connected Autonomous Vehicles (CAVs).

   A V2V approach with FL allows vehicles to share updated knowledge with each other, which is crucial for the safe and efficient operation of vehicle networks. Studies such as that by Lu et al. demonstrate a practical application using Roadside Units (RSUs) act as intermediaries for routing vehicle identities and essential data, facilitating the direct exchange of information between vehicles when necessary.

   This system, in addition to protecting data privacy, helps reduce the risk of leaking information in vehicular cyber-physical environments (VCPS), enabling safe and efficient collaboration in real time.

2. **Mobile Services:**

   Mobile services based on IoT devices are an important application scenario for DFL, making use of the capabilities of smartphones, PCs, among others. These devices are equipped with several types of sensors and cameras that allow them to collect a big range of information sources. Unlike the relatively fixed connectivity of CAVs, mobile IoT devices offer more flexible systems and platforms to support a wide range of applications.

3. **HealthCare:**

   Hospital centers and clinics are beginning to shift from CFL to DFL due to their diverse private data and computing resources. Warnat-Herresthal et al. presented

the Swarm Learning, a DFL framework that deals with 4 heterogeneous disease use cases, including COVID-19, tuberculosis, leukaemia and lung pathology.

The framework incorporates blockchain smart contracts to provide additional security, and dynamically selects leaders to aggregate model parameters at each iteration.

4. **Satellites and Unmanned Aerial Vehicles:**

   UAVs and satellites in dynamic computing environments have vast amounts of sensitive data for remote sensing, target recognition, and military-related tasks under constraints of limited resources. This makes them suited to the benefits of DFL. Bandwidth is a valuable resource for mobile UAVs and satellites. A potential solution is that the application of a DFL framework based on broadcast gossip, customised to dynamic geographical locations, can significantly reduce bandwidth requirements and the consumption of communication resources.

   Additionally, due to its dynamic nature and highly variable data, DFL can also improve real-time responsiveness, adaptability and efficiency of task execution.

2.4 Split Learning

Split learning (SL) has been proposed as a method for enabling resource-constrained devices to train multi-parameter neural networks (NNs) and participate in FL. In essence, Split Learning (SL) divides the Neural Network (NN) model into sections, enabling clients (devices) to transfer the largest section as a processing task to a computationally robust helper. In SL, it is possible to have multiple helpers that can process model parts of one or more clients, resulting in a notable reduction in the maximum training time across all clients.

In the field of collaborative machine learning, SL provides distinct advantages over traditional FL by addressing the resource constraints and privacy challenges of small devices such as IoT sensors and mobile devices. SL allows these devices to participate in training complex models by transferring only a portion of the model to powerful helpers, which perform the more computationally intensive tasks. This model split reduces both the memory footprint and computational demand on client devices, which is essential for integrating learning frameworks in environments with limited resources.

2.4.1 Advantages of Split Learning

SL's architecture inherently supports several key benefits

1. **Resource Efficiency:**

   By offloading the heaviest computations to a helper, SL allows smaller devices to contribute to model training without exhausting their limited resources. This makes SL appropriate for IoT environments and low-processing-power smart devices.

2. **Device Heterogeneity:**

   SL's split structure allows it to work across various hardware configurations and network capabilities. For example, SL´s heterogeneous structure enables the workload related to model training to be distributed among devices, helpers, and cloud, taking advantage of diverse computational resources to support scalable networks.

2.4.2 SL Architecture

In SL, the neural network model is divided into sections, typically at two points called "cut layers". The client device handles the initial layers, which process the raw data. Then, only the activations (or "smashed data") from this initial processing are sent to the server, which takes over the heavier parts of the computation. This division reduces the memory and computational load on the client side and keeps sensitive raw data local, increasing privacy.
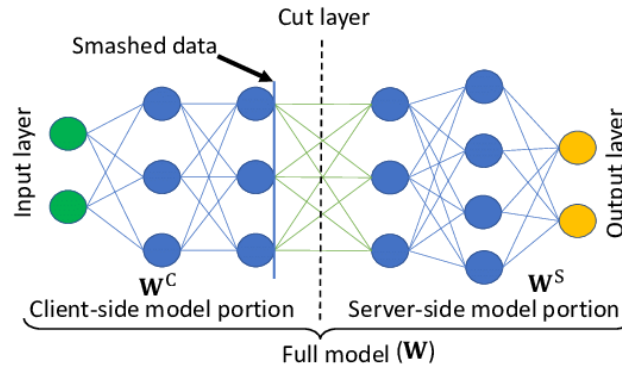
Figure 2.5: Illustration of split learning architecture

Image source: Advancements of federated learning towards privacy preservation: from federated learning to split learning

2.4.3 Applications of Split Learning in Emerging Fields

SL's privacy and resource-efficiency features make it suitable for various domains, including:

1. **Healthcare:**

   In collaborative medical AI, hospitals can share insights from their data without sharing actual data. For instance, in diagnosing conditions from imaging data, only processed features are sent to the helper, allowing each hospital to maintain patient confidentiality. SL enhance this capability by accelerating training across numerous institutions without centralizing data.

Currently, image processing applications in hospital imaging services use this type of mechanism and are already able to detect anomalies and, for example, monitor the development of nodules, comparing old images with current ones.

2. **Smart Cities and 6G Networks:**

SL fits well with the distributed and resource-intensive requirements of future smart cities. SL can support real-time applications such as traffic monitoring and public safety by distributing the computational load across edge servers and mobile devices. In these dynamic configurations, SL's ability to utilize multiple edge nodes in a distributed manner is particularly advantageous.

3. **Autonomous Systems:**

Autonomous vehicles and drones require frequent updates to machine learning models for tasks like object detection. SL enables these devices to offload intensive computations to nearby servers, facilitating real-time updates without overloading on-board systems.

2.5 Delegation protocols (TBD)

1. **Request Helper:**
When a node detects insufficient processing power for training tasks, it sends a helper request to the delegation service.

2. **Helper Assignment:**
The delegation service identifies an optimal helper from the membership list and initiates the Babel-based communication protocol to connect the original node with the helper.

3. **Session Establishment:**
Babel's channel abstraction facilitates creating a communication channel, through which the delegating node and the helper node share training layer information.

....

2.6 Helper membership management (TBD)

1. **Telemetry Monitoring:**
Each node sends telemetry data periodically to the delegation service, which includes CPU, GPU usage, RAM, and battery status.

2. **Candidate Selection:**
Nodes with lower CPU/GPU usage, higher RAM available, and better battery status are prioritized as potential helpers.

16

3. **Membership Protocol Adaptation:**
   Using a gossip or HyParView, can help in maintaining an updated list of potential helper nodes. Babel can support this through its event-driven, message-passing structure for efficient data dissemination

....

2.7 Improvements on Split Learning

SL as already mentioned is a collaborative machine learning approach that partitions a model between clients and a central server, enabling joint training without sharing raw data. This method offers advantages in privacy preservation and resource efficiency. However, it also presents certain challenges and limitations.
Recent advances aim to improve SL by addressing the inherent limitations of the approach.

1. **Multi-Level Split Federated Learning (Split Federated Learning (SFL)):**

   This approach combines the benefits of SL and FL to improve scalability and reduce communication delays. By introducing a multi-level aggregation architecture, it addresses system and statistical heterogeneity in large-scale AIoT systems with non-IID data distributions.

2. **SplitFed Learning without Client-Side Synchronization:**

   Traditional SplitFed Learning requires synchronization among clients, leading to communication and computation overhead. An alternative approach removes this requirement, resulting in a "Multi-head Split Learning" architecture. Empirical studies indicate that this method achieves performance comparable to traditional SplitFed Learning while reducing client-side overhead.

2.7.1 Open Challanges

Several studies have identified challenges in SL that remain unresolved, such as Privacy Leakage which despite SL's design to protect data privacy, research has demonstrated the existence of potential vulnerabilities that could allow adversaries to reconstruct the original data from gradients or shared activations. This suggests that SL may not provide comprehensive protection for sensitive information.

2.7.2 Limitations of Existing Solutions

While enhancements have been proposed, certain limitations remain:

1. **Communication Overhead:**

Some solutions, such as traditional SFL, introduce considerable communication overhead due to the necessity for client-side synchronization. This can prove to be a significant burden for devices with limited resources.

2. **Scalability Issues:**

Despite the fact that multi-level SFL is designed to enhance scalability, it may still face challenges in the context of extremely large and heterogeneous networks, particularly when confronted with non-IID data distributions.

These factors described below, once combined, can make efficient scaling difficult in some environments:

Diverse Data Patterns:

Non-IID data means each client has significantly different data, making it harder to train a unified model that performs well for everyone.

Resource Constraints:

Large networks often involve devices with varying computational power, causing delays and inefficiencies during training.

Communication Bottlenecks:

As the network grows, the amount of data exchanged increases, leading to delays and higher costs in synchronizing updates across devices.

2.7.3 Discussion

Our proposed solution addresses the aforementioned limitations by staying with "normal" SL rather than adopting SFL, due to its significantly simpler development and operational advantages:

1. **Enhances Privacy:**

By keeping the focus on standard Split Learning, our approach reduces the complexity of client-side synchronization, minimizing the risk of additional data leakage through intermediate synchronization steps required in SplitFed Learning.

2. **Reduces Communication Overhead:**

Split Learning inherently involves fewer data exchanges compared to SplitFed Learning, as it avoids the frequent client-to-client synchronization that SFL requires. This results in a more efficient and streamlined communication process.

3. **Simplifies Development:**

The development of SplitFed Learning is notably more complex due to the need to coordinate and synchronize updates across multiple clients. By using SL, our solution maintains a straightforward and more manageable architecture.

By focusing on standard SL, our solution prioritizes simplicity and efficiency while addressing key challenges in collaborative machine learning, making it a robust and practical choice for real-world applications.

2.8 FL frameworks

Machine learning as a service (MLaaS) has increased in recent decades due to increased collection and processing of big data, availability of public APIs, new advanced machine learning (Machine Learning (ML)) methods, open source libraries, tools for ML analysis large-scale and cloud-based computing. These types of systems tended to be predominantly centralized, but with the emergence of Federated Learning due to concerns in terms of privacy, scalability and devices diversity, new services more dedicated to Federated Learning have started to appear, the so called Federated Learning as a Service. Below will be presented some of these services.

1. **TensorFlow Federated (TFF):**

   ...

2. **Flower:**

   ...

3. **PySyft by OpenMined:**

   ...

4. **OpenFL by Intel:**

   ...

5. **FLaaS by Telefónica:**

   ...

2.8.1 FL applications used to train models and classify sets

....

2.8.2 Discussion

....

3

Planning

This Chapter aims at exemplifying how to do common stuff with LaTeX. We also show some stuff which is not that common! ;)

Please, use these examples as a starting point, but you should always consider using the *Big Oracle* (aka, Google, your best friend) to search for additional information or al-ternative ways for achieving similar results.

3.1 Our project

...

3.2 TBD

Citing something online [0, 0, 0].

3.3 TBD

...

3.4 Evaluation Planning

...

3.5 Scheduling

...

Bibliography

[0] C. Artho, K. Havelund, and A. Biere. *High-Level Data Races*. 2003. URL: citeseer.ist.psu.edu/artho03highlevel.html.

[0] C. Artho, K. Havelund, and A. Biere. "Using Block-Local Atomicity to Detect Stale-Value Concurrency Errors". In: *ATVA*. Ed. by F. Wang. Vol. 3299. Lecture Notes in Computer Science. Springer, 2004, pp. 150–164. ISBN: 3-540-23610-4.

[0] N. E. Beckman, K. Bierhoff, and J. Aldrich. "Verifying Correct Usage of Atomic Blocks and Typestate". In: *SIGPLAN Not.* 43.10 (2008), pp. 227–244. ISSN: 0362-1340. DOI: http://doi.acm.org/10.1145/1449955.1449783.

[0] W. contributors. *Shunting-yard algorithm — Wikipedia, The Free Encyclopedia*. 2017-03. URL: https://en.wikipedia.org/w/index.php?title=Shunting-yard_algorithm&oldid=817901155 (visited on 2018-03-01) (cit. on p. 20).

[0] R. J. Dias et al. "Verification of Snapshot Isolation in Transactional Memory Java Programs". In: *Proceedings of the 26th European conference on Object-oriented programming (ECOOP'12)*. Springer-Verlag, 2012-06.

[0] J. Eliot and B. Moss. *Nested transactions: an approach to reliable distributed computing*. Cambridge, MA, USA: Massachusetts Institute of Technology, 1985. ISBN: 0-262-13200-1.

[0] *Fast Lexical Analyser*. URL: https://github.com/westes/flex (visited on 2020-07-26) (cit. on p. 20).

[0] C. Flanagan and S. N. Freund. "Atomizer: a dynamic atomicity checker for multi-threaded programs". In: *POPL '04: Proceedings of the 31st ACM SIGPLAN-SIGACT symposium on Principles of programming languages*. Venice, Italy: ACM, 2004, pp. 256–267. ISBN: 1-58113-729-X. DOI: http://doi.acm.org/10.1145/964001.964023.

[0] *Gnu Bison*. URL: https://www.gnu.org/software/bison/ (visited on 2020-07-26) (cit. on p. 20).

[0] *IBM's Concurrency Testing Repository*.

[0]   N. Shavit and D. Touitou. "Software transactional memory". In: *PODC '95: Proceedings of the fourteenth annual ACM symposium on Principles of distributed computing*. Ottowa, Ontario, Canada: ACM, 1995, pp. 204–213. ISBN: 0-89791-710-3. DOI: http://doi.acm.org/10.1145/224964.224987.

[0]   A. Silberschatz, H. F. Korth, and S. Sudarshan. *Database System Concepts*. Fifth. McGraw-Hill, 2006. ISBN: 007-124476-X.

[0]   C. von Praun and T. R. Gross. "Static Detection of Atomicity Violations in Object-Oriented Programs". In: *Journal of Object Technology*. 2003, p. 2004.

A

NOVAThesis covers showcase

This Appendix shows examples of covers for some of the supported Schools. When the Schools have very similar covers (e.g., all the schools from Universidade do Minho), just one cover is shown. If the covers for MSc dissertations and PhD thesis are considerable different (e.g., for FCT-NOVA and UMinho), then both are shown.

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

Nulla malesuada porttitor diam. Donec felis erat, congue non, volutpat at, tincidunt tristique, libero. Vivamus viverra fermentum felis. Donec nonummy pellentesque ante. Phasellus adipiscing semper elit. Proin fermentum massa ac quam. Sed diam turpis, molestie vitae, placerat a, molestie nec, leo. Maecenas lacinia. Nam ipsum ligula, eleifend at, accumsan nec, suscipit a, ipsum. Morbi blandit ligula feugiat magna. Nunc eleifend consequat lorem. Sed lacinia nulla vitae enim. Pellentesque tincidunt purus vel magna. Integer non enim. Praesent euismod nunc eu purus. Donec bibendum quam in tellus. Nullam cursus pulvinar lectus. Donec et mi. Nam vulputate metus eu enim. Vestibulum

pellentesque felis eu massa.

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.

Fusce mauris. Vestibulum luctus nibh at lectus. Sed bibendum, nulla a faucibus semper, leo velit ultricies tellus, ac venenatis arcu wisi vel nisl. Vestibulum diam. Aliquam pellentesque, augue quis sagittis posuere, turpis lacus congue quam, in hendrerit risus eros eget felis. Maecenas eget erat in sapien mattis porttitor. Vestibulum porttitor. Nulla facilisi. Sed a turpis eu lacus commodo facilisis. Morbi fringilla, wisi in dignissim interdum, justo lectus sagittis dui, et vehicula libero dui cursus dui. Mauris tempor ligula sed lacus. Duis cursus enim ut augue. Cras ac magna. Cras nulla. Nulla egestas. Curabitur a leo. Quisque egestas wisi eget nunc. Nam feugiat lacus vel est. Curabitur consectetuer.

Suspendisse vel felis. Ut lorem lorem, interdum eu, tincidunt sit amet, laoreet vitae, arcu. Aenean faucibus pede eu ante. Praesent enim elit, rutrum at, molestie non, nonummy vel, nisl. Ut lectus eros, malesuada sit amet, fermentum eu, sodales cursus, magna. Donec eu purus. Quisque vehicula, urna sed ultricies auctor, pede lorem egestas dui, et convallis elit erat sed nulla. Donec luctus. Curabitur et nunc. Aliquam dolor odio, commodo pretium, ultricies non, pharetra in, velit. Integer arcu est, nonummy in, fermentum faucibus, egestas vel, odio.

Sed commodo posuere pede. Mauris ut est. Ut quis purus. Sed ac odio. Sed vehicula hendrerit sem. Duis non odio. Morbi ut dui. Sed accumsan risus eget odio. In hac habitasse platea dictumst. Pellentesque non elit. Fusce sed justo eu urna porta tincidunt. Mauris felis odio, sollicitudin sed, volutpat a, ornare ac, erat. Morbi quis dolor. Donec pellentesque, erat ac sagittis semper, nunc dui lobortis purus, quis congue purus metus ultricies tellus. Proin et quam. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos hymenaeos. Praesent sapien turpis, fermentum vel, eleifend faucibus, vehicula eu, lacus.

Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Donec odio elit, dictum in, hendrerit sit amet, egestas sed, leo. Praesent feugiat sapien aliquet odio. Integer vitae justo. Aliquam vestibulum fringilla lorem. Sed neque lectus, consectetuer at, consectetuer sed, eleifend ac, lectus. Nulla facilisi. Pellentesque eget lectus. Proin eu metus. Sed porttitor. In hac habitasse platea dictumst. Suspendisse eu lectus. Ut mi mi, lacinia sit amet, placerat et, mollis vitae, dui. Sed ante tellus, tristique ut, iaculis eu, malesuada ac, dui. Mauris nibh leo, facilisis non, adipiscing quis, ultrices a, dui.

Morbi luctus, wisi viverra faucibus pretium, nibh est placerat odio, nec commodo wisi enim eget quam. Quisque libero justo, consectetuer a, feugiat vitae, porttitor eu, libero. Suspendisse sed mauris vitae elit sollicitudin malesuada. Maecenas ultricies eros

sit amet ante. Ut venenatis velit. Maecenas sed mi eget dui varius euismod. Phasellus aliquet volutpat odio. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Pellentesque sit amet pede ac sem eleifend consectetuer. Nullam elementum, urna vel imperdiet sodales, elit ipsum pharetra ligula, ac pretium ante justo a nulla. Curabitur tristique arcu eu metus. Vestibulum lectus. Proin mauris. Proin eu nunc eu urna hendrerit faucibus. Aliquam auctor, pede consequat laoreet varius, eros tellus scelerisque quam, pellentesque hendrerit ipsum dolor sed augue. Nulla nec lacus.

Suspendisse vitae elit. Aliquam arcu neque, ornare in, ullamcorper quis, commodo eu, libero. Fusce sagittis erat at erat tristique mollis. Maecenas sapien libero, molestie et, lobortis in, sodales eget, dui. Morbi ultrices rutrum lorem. Nam elementum ullamcorper leo. Morbi dui. Aliquam sagittis. Nunc placerat. Pellentesque tristique sodales est. Maecenas imperdiet lacinia velit. Cras non urna. Morbi eros pede, suscipit ac, varius vel, egestas non, eros. Praesent malesuada, diam id pretium elementum, eros sem dictum tortor, vel consectetuer odio sem sed wisi.

Sed feugiat. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Ut pellentesque augue sed urna. Vestibulum diam eros, fringilla et, consectetuer eu, nonummy id, sapien. Nullam at lectus. In sagittis ultrices mauris. Curabitur malesuada erat sit amet massa. Fusce blandit. Aliquam erat volutpat. Aliquam euismod. Aenean vel lectus. Nunc imperdiet justo nec dolor.

Etiam euismod. Fusce facilisis lacinia dui. Suspendisse potenti. In mi erat, cursus id, nonummy sed, ullamcorper eget, sapien. Praesent pretium, magna in eleifend egestas, pede pede pretium lorem, quis consectetuer tortor sapien facilisis magna. Mauris quis magna varius nulla scelerisque imperdiet. Aliquam non quam. Aliquam porttitor quam a lacus. Praesent vel arcu ut tortor cursus volutpat. In vitae pede quis diam bibendum placerat. Fusce elementum convallis neque. Sed dolor orci, scelerisque ac, dapibus nec, ultricies ut, mi. Duis nec dui quis leo sagittis commodo.

Aliquam lectus. Vivamus leo. Quisque ornare tellus ullamcorper nulla. Mauris porttitor pharetra tortor. Sed fringilla justo sed mauris. Mauris tellus. Sed non leo. Nullam elementum, magna in cursus sodales, augue est scelerisque sapien, venenatis congue nulla arcu et pede. Ut suscipit enim vel sapien. Donec congue. Maecenas urna mi, suscipit in, placerat ut, vestibulum ut, massa. Fusce ultrices nulla et nisl.

Etiam ac leo a risus tristique nonummy. Donec dignissim tincidunt nulla. Vestibulum rhoncus molestie odio. Sed lobortis, justo et pretium lobortis, mauris turpis condimentum augue, nec ultricies nibh arcu pretium enim. Nunc purus neque, placerat id, imperdiet sed, pellentesque nec, nisl. Vestibulum imperdiet neque non sem accumsan laoreet. In hac habitasse platea dictumst. Etiam condimentum facilisis libero. Suspendisse in elit quis nisl aliquam dapibus. Pellentesque auctor sapien. Sed egestas sapien nec lectus. Pellentesque vel dui vel neque bibendum viverra. Aliquam porttitor nisl nec pede. Proin mattis libero vel turpis. Donec rutrum mauris et libero. Proin euismod porta felis. Nam lobortis, metus quis elementum commodo, nunc lectus elementum mauris, eget vulputate ligula tellus eu neque. Vivamus eu dolor.

Nulla in ipsum. Praesent eros nulla, congue vitae, euismod ut, commodo a, wisi. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Aenean nonummy magna non leo. Sed felis erat, ullamcorper in, dictum non, ultricies ut, lectus. Proin vel arcu a odio lobortis euismod. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Proin ut est. Aliquam odio. Pellentesque massa turpis, cursus eu, euismod nec, tempor congue, nulla. Duis viverra gravida mauris. Cras tincidunt. Curabitur eros ligula, varius ut, pulvinar in, cursus faucibus, augue.

Nulla mattis luctus nulla. Duis commodo velit at leo. Aliquam vulputate magna et leo. Nam vestibulum ullamcorper leo. Vestibulum condimentum rutrum mauris. Donec id mauris. Morbi molestie justo et pede. Vivamus eget turpis sed nisl cursus tempor. Curabitur mollis sapien condimentum nunc. In wisi nisl, malesuada at, dignissim sit amet, lobortis in, odio. Aenean consequat arcu a ante. Pellentesque porta elit sit amet orci. Etiam at turpis nec elit ultricies imperdiet. Nulla facilisi. In hac habitasse platea dictumst. Suspendisse viverra aliquam risus. Nullam pede justo, molestie nonummy, scelerisque eu, facilisis vel, arcu.

Curabitur tellus magna, porttitor a, commodo a, commodo in, tortor. Donec interdum. Praesent scelerisque. Maecenas posuere sodales odio. Vivamus metus lacus, varius quis, imperdiet quis, rhoncus a, turpis. Etiam ligula arcu, elementum a, venenatis quis, sollicitudin sed, metus. Donec nunc pede, tincidunt in, venenatis vitae, faucibus vel, nibh. Pellentesque wisi. Nullam malesuada. Morbi ut tellus ut pede tincidunt porta. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Etiam congue neque id dolor.

Donec et nisl at wisi luctus bibendum. Nam interdum tellus ac libero. Sed sem justo, laoreet vitae, fringilla at, adipiscing ut, nibh. Maecenas non sem quis tortor eleifend fermentum. Etiam id tortor ac mauris porta vulputate. Integer porta neque vitae massa. Maecenas tempus libero a libero posuere dictum. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Aenean quis mauris sed elit commodo placerat. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos hymenaeos. Vivamus rhoncus tincidunt libero. Etiam elementum pretium justo. Vivamus est. Morbi a tellus eget pede tristique commodo. Nulla nisl. Vestibulum sed nisl eu sapien cursus rutrum.

Nulla non mauris vitae wisi posuere convallis. Sed eu nulla nec eros scelerisque pharetra. Nullam varius. Etiam dignissim elementum metus. Vestibulum faucibus, metus sit amet mattis rhoncus, sapien dui laoreet odio, nec ultricies nibh augue a enim. Fusce in ligula. Quisque at magna et nulla commodo consequat. Proin accumsan imperdiet sem. Nunc porta. Donec feugiat mi at justo. Phasellus facilisis ipsum quis ante. In ac elit eget ipsum pharetra faucibus. Maecenas viverra nulla in massa.

Nulla ac nisl. Nullam urna nulla, ullamcorper in, interdum sit amet, gravida ut, risus. Aenean ac enim. In luctus. Phasellus eu quam vitae turpis viverra pellentesque. Duis feugiat felis ut enim. Phasellus pharetra, sem id porttitor sodales, magna nunc aliquet nibh, nec blandit nisl mauris at pede. Suspendisse risus risus, lobortis eget, semper at,

imperdiet sit amet, quam. Quisque scelerisque dapibus nibh. Nam enim. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Nunc ut metus. Ut metus justo, auctor at, ultrices eu, sagittis ut, purus. Aliquam aliquam.

A.1   A section here

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

Nulla malesuada porttitor diam. Donec felis erat, congue non, volutpat at, tincidunt tristique, libero. Vivamus viverra fermentum felis. Donec nonummy pellentesque ante. Phasellus adipiscing semper elit. Proin fermentum massa ac quam. Sed diam turpis, molestie vitae, placerat a, molestie nec, leo. Maecenas lacinia. Nam ipsum ligula, eleifend at, accumsan nec, suscipit a, ipsum. Morbi blandit ligula feugiat magna. Nunc eleifend consequat lorem. Sed lacinia nulla vitae enim. Pellentesque tincidunt purus vel magna. Integer non enim. Praesent euismod nunc eu purus. Donec bibendum quam in tellus. Nullam cursus pulvinar lectus. Donec et mi. Nam vulputate metus eu enim. Vestibulum pellentesque felis eu massa.

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.

Fusce mauris. Vestibulum luctus nibh at lectus. Sed bibendum, nulla a faucibus semper, leo velit ultricies tellus, ac venenatis arcu wisi vel nisl. Vestibulum diam. Aliquam pellentesque, augue quis sagittis posuere, turpis lacus congue quam, in hendrerit risus

eros eget felis. Maecenas eget erat in sapien mattis porttitor. Vestibulum porttitor. Nulla facilisi. Sed a turpis eu lacus commodo facilisis. Morbi fringilla, wisi in dignissim interdum, justo lectus sagittis dui, et vehicula libero dui cursus dui. Mauris tempor ligula sed lacus. Duis cursus enim ut augue. Cras ac magna. Cras nulla. Nulla egestas. Curabitur a leo. Quisque egestas wisi eget nunc. Nam feugiat lacus vel est. Curabitur consectetuer.

Suspendisse vel felis. Ut lorem lorem, interdum eu, tincidunt sit amet, laoreet vitae, arcu. Aenean faucibus pede eu ante. Praesent enim elit, rutrum at, molestie non, nonummy vel, nisl. Ut lectus eros, malesuada sit amet, fermentum eu, sodales cursus, magna. Donec eu purus. Quisque vehicula, urna sed ultricies auctor, pede lorem egestas dui, et convallis elit erat sed nulla. Donec luctus. Curabitur et nunc. Aliquam dolor odio, commodo pretium, ultricies non, pharetra in, velit. Integer arcu est, nonummy in, fermentum faucibus, egestas vel, odio.

Sed commodo posuere pede. Mauris ut est. Ut quis purus. Sed ac odio. Sed vehicula hendrerit sem. Duis non odio. Morbi ut dui. Sed accumsan risus eget odio. In hac habitasse platea dictumst. Pellentesque non elit. Fusce sed justo eu urna porta tincidunt. Mauris felis odio, sollicitudin sed, volutpat a, ornare ac, erat. Morbi quis dolor. Donec pellentesque, erat ac sagittis semper, nunc dui lobortis purus, quis congue purus metus ultricies tellus. Proin et quam. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos hymenaeos. Praesent sapien turpis, fermentum vel, eleifend faucibus, vehicula eu, lacus.

Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Donec odio elit, dictum in, hendrerit sit amet, egestas sed, leo. Praesent feugiat sapien aliquet odio. Integer vitae justo. Aliquam vestibulum fringilla lorem. Sed neque lectus, consectetuer at, consectetuer sed, eleifend ac, lectus. Nulla facilisi. Pellentesque eget lectus. Proin eu metus. Sed porttitor. In hac habitasse platea dictumst. Suspendisse eu lectus. Ut mi mi, lacinia sit amet, placerat et, mollis vitae, dui. Sed ante tellus, tristique ut, iaculis eu, malesuada ac, dui. Mauris nibh leo, facilisis non, adipiscing quis, ultrices a, dui.

Morbi luctus, wisi viverra faucibus pretium, nibh est placerat odio, nec commodo wisi enim eget quam. Quisque libero justo, consectetuer a, feugiat vitae, porttitor eu, libero. Suspendisse sed mauris vitae elit sollicitudin malesuada. Maecenas ultricies eros sit amet ante. Ut venenatis velit. Maecenas sed mi eget dui varius euismod. Phasellus aliquet volutpat odio. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Pellentesque sit amet pede ac sem eleifend consectetuer. Nullam elementum, urna vel imperdiet sodales, elit ipsum pharetra ligula, ac pretium ante justo a nulla. Curabitur tristique arcu eu metus. Vestibulum lectus. Proin mauris. Proin eu nunc eu urna hendrerit faucibus. Aliquam auctor, pede consequat laoreet varius, eros tellus scelerisque quam, pellentesque hendrerit ipsum dolor sed augue. Nulla nec lacus.

Suspendisse vitae elit. Aliquam arcu neque, ornare in, ullamcorper quis, commodo eu, libero. Fusce sagittis erat at erat tristique mollis. Maecenas sapien libero, molestie et, lobortis in, sodales eget, dui. Morbi ultrices rutrum lorem. Nam elementum ullamcorper

leo. Morbi dui. Aliquam sagittis. Nunc placerat. Pellentesque tristique sodales est. Maecenas imperdiet lacinia velit. Cras non urna. Morbi eros pede, suscipit ac, varius vel, egestas non, eros. Praesent malesuada, diam id pretium elementum, eros sem dictum tortor, vel consectetuer odio sem sed wisi.

Sed feugiat. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Ut pellentesque augue sed urna. Vestibulum diam eros, fringilla et, consectetuer eu, nonummy id, sapien. Nullam at lectus. In sagittis ultrices mauris. Curabitur malesuada erat sit amet massa. Fusce blandit. Aliquam erat volutpat. Aliquam euismod. Aenean vel lectus. Nunc imperdiet justo nec dolor.

Etiam euismod. Fusce facilisis lacinia dui. Suspendisse potenti. In mi erat, cursus id, nonummy sed, ullamcorper eget, sapien. Praesent pretium, magna in eleifend egestas, pede pede pretium lorem, quis consectetuer tortor sapien facilisis magna. Mauris quis magna varius nulla scelerisque imperdiet. Aliquam non quam. Aliquam porttitor quam a lacus. Praesent vel arcu ut tortor cursus volutpat. In vitae pede quis diam bibendum placerat. Fusce elementum convallis neque. Sed dolor orci, scelerisque ac, dapibus nec, ultricies ut, mi. Duis nec dui quis leo sagittis commodo.

Aliquam lectus. Vivamus leo. Quisque ornare tellus ullamcorper nulla. Mauris porttitor pharetra tortor. Sed fringilla justo sed mauris. Mauris tellus. Sed non leo. Nullam elementum, magna in cursus sodales, augue est scelerisque sapien, venenatis congue nulla arcu et pede. Ut suscipit enim vel sapien. Donec congue. Maecenas urna mi, suscipit in, placerat ut, vestibulum ut, massa. Fusce ultrices nulla et nisl.

Etiam ac leo a risus tristique nonummy. Donec dignissim tincidunt nulla. Vestibulum rhoncus molestie odio. Sed lobortis, justo et pretium lobortis, mauris turpis condimentum augue, nec ultricies nibh arcu pretium enim. Nunc purus neque, placerat id, imperdiet sed, pellentesque nec, nisl. Vestibulum imperdiet neque non sem accumsan laoreet. In hac habitasse platea dictumst. Etiam condimentum facilisis libero. Suspendisse in elit quis nisl aliquam dapibus. Pellentesque auctor sapien. Sed egestas sapien nec lectus. Pellentesque vel dui vel neque bibendum viverra. Aliquam porttitor nisl nec pede. Proin mattis libero vel turpis. Donec rutrum mauris et libero. Proin euismod porta felis. Nam lobortis, metus quis elementum commodo, nunc lectus elementum mauris, eget vulputate ligula tellus eu neque. Vivamus eu dolor.

Nulla in ipsum. Praesent eros nulla, congue vitae, euismod ut, commodo a, wisi. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Aenean nonummy magna non leo. Sed felis erat, ullamcorper in, dictum non, ultricies ut, lectus. Proin vel arcu a odio lobortis euismod. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Proin ut est. Aliquam odio. Pellentesque massa turpis, cursus eu, euismod nec, tempor congue, nulla. Duis viverra gravida mauris. Cras tincidunt. Curabitur eros ligula, varius ut, pulvinar in, cursus faucibus, augue.

Nulla mattis luctus nulla. Duis commodo velit at leo. Aliquam vulputate magna et leo. Nam vestibulum ullamcorper leo. Vestibulum condimentum rutrum mauris. Donec

id mauris. Morbi molestie justo et pede. Vivamus eget turpis sed nisl cursus tempor. Curabitur mollis sapien condimentum nunc. In wisi nisl, malesuada at, dignissim sit amet, lobortis in, odio. Aenean consequat arcu a ante. Pellentesque porta elit sit amet orci. Etiam at turpis nec elit ultricies imperdiet. Nulla facilisi. In hac habitasse platea dictumst. Suspendisse viverra aliquam risus. Nullam pede justo, molestie nonummy, scelerisque eu, facilisis vel, arcu.

Curabitur tellus magna, porttitor a, commodo a, commodo in, tortor. Donec interdum. Praesent scelerisque. Maecenas posuere sodales odio. Vivamus metus lacus, varius quis, imperdiet quis, rhoncus a, turpis. Etiam ligula arcu, elementum a, venenatis quis, sollicitudin sed, metus. Donec nunc pede, tincidunt in, venenatis vitae, faucibus vel, nibh. Pellentesque wisi. Nullam malesuada. Morbi ut tellus ut pede tincidunt porta. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Etiam congue neque id dolor.

Donec et nisl at wisi luctus bibendum. Nam interdum tellus ac libero. Sed sem justo, laoreet vitae, fringilla at, adipiscing ut, nibh. Maecenas non sem quis tortor eleifend fermentum. Etiam id tortor ac mauris porta vulputate. Integer porta neque vitae massa. Maecenas tempus libero a libero posuere dictum. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Aenean quis mauris sed elit commodo placerat. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos hymenaeos. Vivamus rhoncus tincidunt libero. Etiam elementum pretium justo. Vivamus est. Morbi a tellus eget pede tristique commodo. Nulla nisl. Vestibulum sed nisl eu sapien cursus rutrum.

Nulla non mauris vitae wisi posuere convallis. Sed eu nulla nec eros scelerisque pharetra. Nullam varius. Etiam dignissim elementum metus. Vestibulum faucibus, metus sit amet mattis rhoncus, sapien dui laoreet odio, nec ultricies nibh augue a enim. Fusce in ligula. Quisque at magna et nulla commodo consequat. Proin accumsan imperdiet sem. Nunc porta. Donec feugiat mi at justo. Phasellus facilisis ipsum quis ante. In ac elit eget ipsum pharetra faucibus. Maecenas viverra nulla in massa.

Nulla ac nisl. Nullam urna nulla, ullamcorper in, interdum sit amet, gravida ut, risus. Aenean ac enim. In luctus. Phasellus eu quam vitae turpis viverra pellentesque. Duis feugiat felis ut enim. Phasellus pharetra, sem id porttitor sodales, magna nunc aliquet nibh, nec blandit nisl mauris at pede. Suspendisse risus risus, lobortis eget, semper at, imperdiet sit amet, quam. Quisque scelerisque dapibus nibh. Nam enim. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Nunc ut metus. Ut metus justo, auctor at, ultrices eu, sagittis ut, purus. Aliquam aliquam.

I

Annex 1 Lorem Ipsum

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

Nulla malesuada porttitor diam. Donec felis erat, congue non, volutpat at, tincidunt tristique, libero. Vivamus viverra fermentum felis. Donec nonummy pellentesque ante. Phasellus adipiscing semper elit. Proin fermentum massa ac quam. Sed diam turpis, molestie vitae, placerat a, molestie nec, leo. Maecenas lacinia. Nam ipsum ligula, eleifend at, accumsan nec, suscipit a, ipsum. Morbi blandit ligula feugiat magna. Nunc eleifend consequat lorem. Sed lacinia nulla vitae enim. Pellentesque tincidunt purus vel magna. Integer non enim. Praesent euismod nunc eu purus. Donec bibendum quam in tellus. Nullam cursus pulvinar lectus. Donec et mi. Nam vulputate metus eu enim. Vestibulum pellentesque felis eu massa.

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum

wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.

Fusce mauris. Vestibulum luctus nibh at lectus. Sed bibendum, nulla a faucibus semper, leo velit ultricies tellus, ac venenatis arcu wisi vel nisl. Vestibulum diam. Aliquam pellentesque, augue quis sagittis posuere, turpis lacus congue quam, in hendrerit risus eros eget felis. Maecenas eget erat in sapien mattis porttitor. Vestibulum porttitor. Nulla facilisi. Sed a turpis eu lacus commodo facilisis. Morbi fringilla, wisi in dignissim interdum, justo lectus sagittis dui, et vehicula libero dui cursus dui. Mauris tempor ligula sed lacus. Duis cursus enim ut augue. Cras ac magna. Cras nulla. Nulla egestas. Curabitur a leo. Quisque egestas wisi eget nunc. Nam feugiat lacus vel est. Curabitur consectetuer.