



UNIVERSIDAD
DE MURCIA

Escuela
de Doctorado

TESIS DOCTORAL

Aprendizaje Federado Descentralizado y Robusto en Entornos Heterogéneos

*Robust Decentralized
Federated Learning in
Heterogeneous Environments*

AUTOR/A

Enrique Tomás Martínez Beltrán

DIRECTOR/ES

Alberto Huertas Celdrán
Gregorio Martínez Pérez

2026



UNIVERSIDAD
DE MURCIA

Escuela
de Doctorado

TESIS DOCTORAL

Aprendizaje Federado Descentralizado y Robusto en Entornos Heterogéneos

*Robust Decentralized
Federated Learning in
Heterogeneous Environments*

AUTOR/A

Enrique Tomás Martínez Beltrán

DIRECTOR/ES

Alberto Huertas Celdrán
Gregorio Martínez Pérez

2026



DECLARACIÓN DE AUTORÍA Y ORIGINALIDAD DE LA TESIS PRESENTADA EN MODALIDAD DE COMPENDIO O ARTÍCULOS PARA OBTENER EL TÍTULO DE DOCTOR/A

Aprobado por la Comisión General de Doctorado el 19 de octubre de 2022.

Yo, D. Enrique Tomás Martínez Beltrán, habiendo cursado el Programa de Doctorado en Informática de la Escuela Internacional de Doctorado de la Universidad de Murcia (EIDUM), como autor/a de la tesis presentada para la obtención del título de Doctor/a titulada:

Robust Decentralized Federated Learning in Heterogeneous Environments / Aprendizaje Federado Descentralizado y Robusto en Entornos Heterogéneos

y dirigida por:

D.: Alberto Huertas Celdrán

D.: Gregorio Martínez Pérez

DECLARO QUE:

La tesis es una obra original que no infringe los derechos de propiedad intelectual ni los derechos de propiedad industrial u otros, de acuerdo con el ordenamiento jurídico vigente, en particular, la Ley de Propiedad Intelectual (R.D. legislativo 1/1996, de 12 de abril, por el que se aprueba el texto refundido de la Ley de Propiedad Intelectual, modificado por la Ley 2/2019, de 1 de marzo, regularizando, aclarando y armonizando las disposiciones legales vigentes sobre la materia), en particular, las disposiciones referidas al derecho de cita, cuando se han utilizado sus resultados o publicaciones.

Además, al haber sido autorizada como prevé el artículo 29.8 del reglamento cuenta con:

- La aceptación por escrito de los coautores de las publicaciones de que el doctorando las presente como parte de la tesis.
- En su caso, la renuncia por escrito de los coautores no doctores de dichos trabajos a presentarlos como parte de otras tesis doctorales en la Universidad de Murcia o en cualquier otra universidad.

Del mismo modo, asumo ante la Universidad cualquier responsabilidad que pudiera derivarse de la autoría o falta de originalidad del contenido de la tesis presentada, en caso de plagio, de conformidad con el ordenamiento jurídico vigente.

Murcia, a 16 de junio de 2026

Fdo.: Enrique Tomás Martínez Beltrán

Información básica sobre protección de sus datos personales aportados:	
Responsable	Universidad de Murcia. Avenida teniente Flomesta, 5. Edificio de la Convalecencia. 30003; Murcia. Delegado de Protección de Datos: dpd@um.es
Legitimación	La Universidad de Murcia se encuentra legitimada para el tratamiento de sus datos por ser necesario para el cumplimiento de una obligación legal aplicable al responsable del tratamiento. art. 6.1.c) del Reglamento General de Protección de Datos
Finalidad	Gestionar su declaración de autoría y originalidad
Destinatarios	No se prevén comunicaciones de datos
Derechos	Los interesados pueden ejercer sus derechos de acceso, rectificación, cancelación, oposición, limitación del tratamiento, olvido y portabilidad a través del procedimiento establecido a tal efecto en el Registro Electrónico o mediante la presentación de la correspondiente solicitud en las Oficinas de Asistencia en Materia de Registro de la Universidad de Murcia

Esta DECLARACIÓN DE AUTORÍA Y ORIGINALIDAD debe ser insertada en la quinta hoja, después de la portada de la tesis presentada para la obtención del título de Doctor/a.

The following PhD Thesis is a compilation of the next published articles, being the PhD candidate the main author in all of them:

- Enrique Tomás Martínez Beltrán, Mario Quiles Pérez, Pedro Miguel Sánchez Sánchez, Sergio López Bernal, G r me Bovet, Manuel Gil P rez, Gregorio Mart nez P rez, Alberto Huertas Celdr n. “**Decentralized Federated Learning: Fundamentals, State of the Art, Frameworks, Trends, and Challenges**”, *IEEE Communications Surveys and Tutorials*, vol. 25, no. 4, pp. 2983-3013, 2023.
DOI: 10.1109/COMST.2023.3315746
JIF 2023: 34.4 (Q1-D1)
- Enrique Tom s Mart nez Beltr n, Pedro Miguel S nchez S nchez, G r me Bovet, Burkhard Stiller, Gregorio Mart nez P rez, Alberto Huertas Celdr n. “**Flighter: Decentralized Federated Learning and Situational Awareness for Secure Military Aerial Reconnaissance**”, *IEEE Communications Magazine*, vol. 63, no. 10, pp. 136-142, 2025.
DOI: 10.1109/MCOM.003.2400401
JIF 2024: 8.3 (Q1-D1)
- Enrique Tom s Mart nez Beltr n, G r me Bovet, Gregorio Mart nez P rez, Alberto Huertas Celdr n. “**Decentralized Federated Learning with Multimodal Prototypes for Heterogeneous Data**”, *Information Fusion*, vol. 132, art. no. 104251, 2026.
DOI: 10.1016/j.inffus.2026.104251
JIF 2024: 15.5 (Q1-D1)

Contents

Acknowledgements	iii
Agradecimientos	v
Abstract	vii
1 Introduction and motivation	vii
2 Objectives	xi
3 Methodology	xii
4 Results	xiv
5 Conclusions and future work	.xviii
Resumen	xxi
1 Introducción y motivación	xxi
2 Objetivos	.xxvi
3 Metodología	xxvii
4 Resultados	.xxix
5 Conclusiones y trabajo futuro	xxxiii
Bibliography	xl
Other relevant works	xli
Publications composing the PhD Thesis	
1 Decentralized Federated Learning: Fundamentals, State of the Art, Frameworks, Trends, and Challenges	3
2 Flighter: Decentralized Federated Learning and Situational Awareness for Secure Military Aerial Reconnaissance	37
3 Decentralized Federated Learning with Multimodal Prototypes for Heterogeneous Data	47

Acknowledgements

To my parents and my sister,

because much of who I am begins with you. Thank you for always being there, even when this path was difficult to explain, to follow, or to understand from the outside. Thank you for your patience, your trust, your quiet support, and for reminding me, each in your own way, that what mattered was not only to reach the end, but to remain myself throughout the journey.

To my partner,

for accompanying me through a stage filled with excitement, exhaustion, uncertainty, pressure, and many long days. Thank you for supporting me when I doubted myself, for celebrating every small step forward with me, and for being present in the less visible parts of this process as well. Thank you for your patience, your love, and for making even the most difficult moments easier to bear.

To my friends,

for being a refuge, a pause, and a reality beyond the thesis. Thank you for the conversations, the trips, the jokes, the improvised plans, and also for understanding my absences. Thank you for reminding me that life continued out there, for sincerely sharing in each achievement, and for staying close even when I was absorbed by this work.

To my “three” thesis supervisors,

for your guidance, dedication, and trust throughout these years. Thank you for opening doors to CyberDataLab for me, for helping me grow as a researcher, for demanding more from me when it was necessary, and for teaching me to approach scientific work with greater rigor, perspective, and ambition. This thesis would not have been possible without your academic and human support, nor without the opportunities, discussions, and lessons shared along this path.

To those who are no longer here,

but who remain part of me. To the people I would have liked to see arrive at this moment and who, in one way or another, are still present in my memory. I wish you could be here to see it.

And to myself,

for having continued. For not giving up when doing so seemed easier, for learning to live with doubt, for accepting mistakes, and for carrying on even in the moments when the end felt too far away. This thesis brings an important stage to a close, but, above all, it remembers everything it took to build it.

To all those who, in one way or another, have been part of this journey: thank you.

Agradecimientos

A mis padres y a mi hermana,

porque buena parte de lo que soy empieza en vosotros. Gracias por haber estado siempre, incluso cuando este camino era difícil de explicar, de seguir o de entender desde fuera. Gracias por la paciencia, por la confianza, por el apoyo silencioso y por recordarme, cada uno a vuestra manera, que lo importante no era solo llegar, sino seguir siendo yo durante el trayecto.

A mi pareja,

por acompañarme en una etapa que ha tenido ilusión, cansancio, incertidumbre, presión y muchos días largos. Gracias por sostenerme cuando yo dudaba, por celebrar conmigo cada pequeño avance y por estar también en las partes menos visibles de este proceso. Gracias por tu paciencia, por tu amor y por haber hecho que incluso los momentos más complicados fueran más llevaderos.

A mis amigos,

por haber sido refugio, pausa y realidad fuera de la tesis. Gracias por las conversaciones, los viajes, las bromas, los planes improvisados y también por entender mis ausencias. Gracias por recordarme que la vida seguía ahí fuera, por alegraros sinceramente de cada logro y por estar cerca incluso cuando yo estaba absorbido por este trabajo.

A mis “tres” directores de tesis,

por su guía, dedicación y confianza durante estos años. Gracias por abrirme las puertas en el CyberDataLab, por ayudarme a crecer como investigador, por exigirme cuando era necesario y por enseñarme a mirar el trabajo científico con más rigor, perspectiva y ambición. Esta tesis no habría sido posible sin vuestro acompañamiento académico y humano, ni sin las oportunidades, discusiones y aprendizajes compartidos a lo largo de este camino.

A quienes ya no están,

pero siguen formando parte de mí. A las personas que me habría gustado ver llegar hasta este momento y que, de una forma u otra, siguen presentes en mi memoria. Ojalá pudierais estar aquí para verlo.

Y a mí mismo,

por haber seguido. Por no rendirme cuando parecía más fácil hacerlo, por aprender a convivir con la duda, por aceptar los errores y por continuar incluso en los momentos en los que el final parecía demasiado lejos. Esta tesis cierra una etapa importante, pero, sobre todo, recuerda todo lo que ha costado construirla.

A todos los que, de una manera u otra, habéis formado parte de este camino: gracias.

1 Introduction and motivation

The increasing distribution of data, computation, and decision-making across modern digital infrastructures has transformed the design, training, and deployment of Machine Learning (ML) models. Current networked environments are no longer composed only of isolated systems or centralized data repositories. They increasingly involve multiple participants, organizations, devices, sensors, and edge nodes that generate and process information locally. This evolution is particularly relevant in the Internet of Things (IoT) and other distributed environments, where data are generated, processed, and governed by multiple participants across industrial, healthcare, vehicular, tactical, and cyber-physical settings, among others. In these scenarios, data are naturally distributed, and centralization may be impractical, costly, or undesirable due to technical, regulatory, privacy-related, or operational constraints [1, 2]. In this context, the ability to extract knowledge from distributed information sources is essential to support data-driven monitoring, automation, and decision-making processes. However, participant diversity and environmental dynamism increase the complexity of building learning systems that are reliable, efficient, and compatible with local data ownership. These constraints motivate learning mechanisms that operate close to the data generation process, while still enabling participants to benefit from collaborative model training.

Artificial Intelligence (AI), and particularly ML, has become a fundamental mechanism for exploiting the value of distributed data. Nevertheless, conventional ML approaches usually assume that data can be collected, transferred, and processed on a centralized infrastructure. This assumption is increasingly difficult to maintain when participants are geographically distributed, data may be sensitive, devices operate with limited computational or energy resources, and communication links are unstable or intermittent [3]. In such cases, transferring all local data to a central location may impose technical and organizational constraints. It can increase communication and latency costs, expose sensitive information, and create governance or operational dependencies on a single infrastructure [4, 5]. These limitations are more pronounced when data are generated continuously at the network edge or when participants belong to different administrative domains. Accordingly, the problem shifts from training accurate models in isolation to designing collaborative learning mechanisms that exploit local information without assuming that all data can be aggregated at a single point for processing.

Federated Learning (FL), originally proposed by Google in 2016, emerged as a collab-

orative learning paradigm designed to support the training of privacy-preserving models without the need to store raw data at a single central point [6]. In FL, participants train models with their local data and exchange model-related information rather than raw datasets. This approach enables the development of shared learning capabilities while keeping data locally stored, which is particularly relevant for distributed and privacy-aware ML. The most common FL configuration has traditionally followed a centralized architecture, where a server coordinates the training process, receives local model updates, aggregates them, and redistributes the resulting global model [4]. Consequently, Centralized Federated Learning (CFL) has become a widely studied approach, with extensive work on optimization, communication, privacy, personalization, robustness, and system design across multiple application domains [5]. Nevertheless, the effectiveness of CFL depends on the coordinating entity being able to manage communication, aggregation, synchronization, and trust-related assumptions throughout the training process.

Although its advantages make it suitable under certain circumstances, CFL introduces structural limitations that become critical in several real-world scenarios. The central server may become a bottleneck as the number of participants grows or as the size of the exchanged models increases. It may also become a single point of failure when the coordinating entity is interrupted, overloaded, attacked, or disconnected [7, 8]. In addition, centralized coordination concentrates trust in a single entity, which may be unsuitable for environments that comprise autonomous participants, mobile nodes, different administrative domains, or infrastructures where no stable and universally trusted coordinator can be assumed [9]. These limitations are particularly evident in distributed, mobile, and tactical environments, where communication links change over time, and participants may need to operate under partial connectivity or operational constraints. Under these conditions, the server-based architecture of CFL can disrupt the continuity of the learning process and reduce the ability of the system to adapt to changes in topology, resources, or availability. These limitations justify the research on alternative forms of collaborative learning architectures in which aggregation and coordination responsibilities are not concentrated in a single entity.

Decentralized Federated Learning (DFL) arises in this context as a paradigm in which participants collaborate without relying on a central aggregation server. Instead of sending updates to a single coordinator, participants exchange information directly with other nodes through peer-to-peer communication, neighbor-based aggregation, consensus mechanisms, or related decentralized coordination strategies [7, 10]. This decentralization can reduce dependence on a given central infrastructure and may improve fault tolerance, scalability, and resilience in scenarios with intermittent connectivity or where the existence of a trusted central server cannot be guaranteed [8, 9]. DFL is therefore especially valuable when autonomous participants need to collaborate under distributed, resource-constrained, and intermittently connected conditions [1]. However, decentralization also modifies the assumptions of the learning process. Participants may have only partial knowledge of the federation, communicate with a limited set of neighbors, and make aggregation decisions from locally available information. DFL is consequently not only an architectural variation of CFL, but a distinct collaborative learning paradigm with its own system, communication, optimization, and security challenges.

The absence of a central coordinator changes the nature of the learning problem. In CFL, responsibilities such as aggregation, synchronization, participant coordination, reliability assessment, and, in some cases, security monitoring are typically handled by the central server. In DFL, these responsibilities must instead be distributed among the participating nodes [10]. This redistribution introduces new challenges related to communication management, topology design, convergence, robustness, trust distribution,

privacy, security, and performance evaluation [11]. The design of DFL systems requires careful consideration of aspects such as coordination, communication, aggregation, reliability assessment, evaluation criteria, and the types of information exchanged among participants. In addition, the communication structure itself becomes part of the learning mechanism, since the topology determines how knowledge propagates throughout the federation and how local biases can affect neighboring participants. For this reason, the value of DFL lies not merely in eliminating a central server, but in providing mechanisms that enable decentralized participants to collaborate reliably under realistic conditions.

A first implication of this redistribution concerns reliability and robustness. Since DFL participants make communication and aggregation decisions based on locally available information, the system must preserve collaborative learning even when some nodes are unreliable, communication conditions vary, or the updates exchanged may be affected by adversarial behavior. Participants may be affected by communication failures, mobility, operational disturbances, or compromised behavior [12]. In centralized architectures, a server may support monitoring, filtering, or coordination tasks. However, in DFL, there is no central authority with a global view of the federation, and the attack surface is redistributed across the communication and aggregation processes [13, 14]. This lack of centralized oversight increases the need for decentralized mechanisms capable of estimating the reliability of exchanged information, reducing the influence of potentially harmful contributions, and preserving collaborative learning under adverse conditions [15, 16]. As a result, robustness in DFL depends not only on the learning algorithm but also on the interaction between topology, communication, participant behavior, and the criteria used to accept, weight, or discard information. The distributed nature of the federation amplifies these issues because a compromised or unreliable participant can influence its neighbors and subsequently affect other parts of the network.

In dynamic and adversarial DFL environments, robustness cannot be assessed only from the consistency of model updates. The reliability of a contribution may also depend on the operational context in which it is generated, including communication stability, resource availability, mobility, sensing conditions, and potentially compromised behavior. These factors are particularly relevant because existing DFL approaches still provide limited support for scenarios in which mobility, unreliable communications, adversarial behavior, and decentralized aggregation must be handled jointly [17]. Situational awareness, traditionally understood as the perception, comprehension, and projection of relevant elements in a changing environment, provides a conceptual basis to connect model-level evidence with contextual information in decentralized collaboration [18]. From this perspective, robust DFL requires reliability-aware mechanisms that evaluate exchanged information not only by its learning behavior, but also by the conditions under which participants interact. Tactical and defense-oriented scenarios provide a demanding example of these conditions, as they combine decentralized collaboration with mobility, constrained communication, operational uncertainty, and potentially adversarial behavior [19]. Within this category, military aerial reconnaissance is particularly relevant because UAVs and other military aerial vehicles may need to jointly perform sensing, computation, communication, and learning while changing position, operating under intermittent connectivity, and facing hostile actions that affect both mission conditions and the collaborative training process [20, 21]. It therefore provides a bounded but demanding setting in which the reliability of DFL can be examined from both model-related and operational perspectives.

A second major challenge concerns heterogeneity, which affects both the participating nodes and the data they hold. Real decentralized environments are rarely composed of participants with identical data, resources, connectivity, or operational conditions. Devices

may differ in resources, connectivity, mobility, availability of local data, and participation patterns over time [22]. Their datasets may also follow non-Independent and Identically Distributed (non-IID) distributions, which means that participants may observe different classes, feature distributions, label proportions, or local data patterns [11, 23]. These forms of heterogeneity can affect training stability, aggregation quality, convergence behavior, and final model performance [24, 22]. In CFL, a central server can coordinate participation, apply global aggregation rules, or monitor local update behavior. In DFL, these responsibilities are distributed, and the effect of heterogeneity may propagate through local neighborhoods. For this reason, DFL requires mechanisms that can operate under unequal resource conditions, skewed data distributions, and partial visibility into the global federation.

Heterogeneity becomes even more complex when learning scenarios involve multiple data modalities. Many applications do not rely on a single source of information, but combine images, audio, text, sensor measurements, telemetry, structured signals, or other forms of data [25, 26]. In decentralized settings, not all participants necessarily have access to the same modalities. Some clients may observe only one modality, others may have complete multimodal samples, and others may experience missing modalities in part of their local data [27]. This situation creates a significant challenge because knowledge must be exchanged and fused without assuming homogeneous data availability across participants. When modalities are missing or unevenly distributed, conventional mechanisms based on the exchange of full model parameters may become inefficient or insufficiently adapted to the structure of the problem [28, 29]. Furthermore, multimodal learning requires not only aggregation, but also representation alignment, modality fusion, and the ability to preserve discriminative information between clients with different local observations. For this reason, multimodal DFL requires communication and learning strategies able to represent useful information even when local data are incomplete, statistically unbalanced, and distributed among participants with different sensing capabilities [30]. This requirement motivates representation-level communication mechanisms that do not rely exclusively on exchanging complete model parameters or gradients.

Prototype-based learning has gained attention in this context because compact, class-level, or modality-aware representations can facilitate knowledge sharing while reducing the communication burden associated with high-dimensional model updates. In CFL, prototype exchange has been used to mitigate data heterogeneity by sharing abstract class representations rather than full gradients or parameters [31, 32]. In decentralized settings, prototype sharing has also been explored to reduce communication overhead and improve model consistency between neighboring clients [33]. For multimodal FL, cross-modal prototype mechanisms have been proposed to address severe missing-modality conditions and representation misalignment [34], while partial-modality missing has motivated specific aggregation and regularization strategies in federated systems [35]. These developments enable heterogeneous data integration, representation-level knowledge exchange, and communication-efficient collaboration without requiring the systematic exchange of complete model parameters. They also illustrate a broader research direction in which information exchanged during federation may consist of compact representations encoding discriminative knowledge. This perspective is central to DFL, where communication occurs among participants, and reducing network consumption may be necessary to maintain collaboration under resource-constrained conditions.

Overall, the literature suggests that DFL is a relevant paradigm for collaborative learning in distributed environments, but its practical adoption remains dependent on a broader set of foundational, methodological, and operational challenges. First, the field needs a systematic understanding of its fundamentals, including architectures, topologies,

communication mechanisms, robustness, security and privacy approaches, key performance indicators, frameworks, application scenarios, trends, and open challenges [7, 11]. Second, decentralized learning must be made more robust to adversarial and dynamic conditions, in which there is no central authority with a complete view of the federation, and harmful behavior may affect both communication and aggregation [15]. Third, DFL systems must address statistical, resource, communication, and multimodal heterogeneity, particularly when clients operate with non-IID data distributions, incomplete modality availability, or different communication capacities [25]. Finally, communication-efficient mechanisms are needed to make DFL viable in resource-constrained settings where full model exchange may be costly or unstable.

Based on these considerations, this PhD thesis investigates how DFL can be advanced toward robust, adaptive, and communication-efficient collaborative learning in heterogeneous environments. It studies DFL as a paradigm that reduces dependence on a central aggregation entity while redistributing the responsibilities for communication, trust, reliability, robustness, and performance assessment among the participating nodes. Within the broader DFL research landscape, the thesis focuses on a coherent subset of problems: the foundational characterization of the field, decentralized reliability under bounded adversarial and dynamic conditions, and communication-efficient learning under statistical and multimodal heterogeneity, including non-IID distributions and missing modalities. From this perspective, the thesis is guided by the following research questions:

- **RQ1:** What are the fundamentals, frameworks, application scenarios, metrics, limitations, and challenges that define the current state of DFL?
- **RQ2:** How can the reliability and robustness of DFL be reinforced against adversarial behavior, operational perturbations, and potentially compromised nodes?
- **RQ3:** What mechanisms can support decentralized multimodal learning under non-IID data and missing-modality conditions while preserving collaborative learning performance?
- **RQ4:** How do communication-efficient mechanisms affect communication cost and collaborative learning performance in DFL under heterogeneous multimodal settings?

2 Objectives

The main goal of this PhD thesis is to advance DFL as a paradigm for robust, adaptive, and communication-efficient collaborative learning in heterogeneous environments. More specifically, the thesis examines DFL as a decentralized learning approach in which participants collaborate without a central aggregator while facing heterogeneous data, changing communication conditions, and potentially adversarial behavior.

The thesis addresses three complementary aspects of DFL, organized as a conceptual progression from foundational characterization to targeted technical responses. First, it analyzes the fundamentals of DFL, including architectures, topologies, communication mechanisms, security and privacy approaches, performance indicators, frameworks, application scenarios, trends, and open challenges. This foundational analysis identifies the challenges posed by decentralization and motivates the need for mechanisms that support reliability, robustness, heterogeneity management, and communication efficiency. On this basis, the thesis articulates two complementary technical lines derived from the challenges identified in the state of the art. The first line examines the adversarial dimension of DFL

by studying how situational awareness can support decentralized reliability in dynamic scenarios affected by operational perturbations, communication attacks, and attacks on the collaborative learning process. The second line addresses statistical and multimodal heterogeneity through prototype-based communication mechanisms, with particular attention to non-IID class distributions, missing modalities, and communication-efficiency constraints. The following specific objectives define the scientific scope of the thesis and indicate their corresponding research questions:

- O.1.** Analyze the current state of the art regarding DFL through a comprehensive review, analysis, and comparison of its fundamentals, architectures, topologies, communication mechanisms, security and privacy approaches, key performance indicators, optimization mechanisms, frameworks, and application scenarios (**RQ1**).
- O.2.** Building on the characterization established in **O.1**, identify and critically assess the main limitations, research gaps, trends, and open challenges in DFL that reveal the need for robust, adaptive, and communication-efficient mechanisms in decentralized federated environments (**RQ1**).
- O.3.** Design and evaluate a technical solution for DFL in dynamic and adversarial scenarios, considering contextual and operational information to support decentralized reliability (**RQ2**).
- O.4.** Assess the robustness of the proposed solution derived from **O.3** against operational perturbations, communication attacks, and attacks affecting the collaborative learning process in a bounded adversarial scenario (**RQ2**).
- O.5.** Design and evaluate a prototype-based technical solution for DFL with heterogeneous multimodal data, considering non-IID class distributions and missing modalities across participants (**RQ3**).
- O.6.** Evaluate the communication efficiency and collaborative learning performance of multimodal prototype exchange within the solution proposed in **O.5**, in comparison with approaches based on the exchange of full model parameters or complete model updates (**RQ4**).

3 Methodology

This PhD thesis was conducted using a scientific approach that combined a comprehensive review of the state of the art, the design of technical solutions, and the evaluation of results across successive research stages. Since the thesis is presented as a compendium of publications, its methodology is articulated through three journal articles that examine the same overarching problem from complementary angles. The first article provides a foundational characterization of DFL and establishes the diagnostic basis for the thesis by identifying the main limitations, trends, and open challenges of the field. From this analysis, two particularly relevant challenges were selected for further technical development: decentralized reliability under adversarial and dynamic conditions, and collaborative learning under statistical and multimodal heterogeneity. Together, these publications form a coherent methodological sequence that supports the common objective of advancing robust, adaptive, and communication-efficient DFL in heterogeneous environments.

The first stage addressed **RQ1** and objectives **O.1** and **O.2** by providing a structured view of DFL as an emerging research paradigm. A comprehensive review of the literature was conducted to identify the aspects that distinguish DFL from CFL, including

federation architectures, network topologies, communication mechanisms, security and privacy approaches, key performance indicators, optimization mechanisms, frameworks, and application scenarios. The reviewed literature was categorized and analyzed to establish the conceptual structure of the field, identify commonly used mechanisms, compare available frameworks, and determine the most relevant trends, lessons learned, and open challenges. This analysis provided the terminology, taxonomy, and analytical basis required to understand why DFL needs solutions specifically designed for decentralized, dynamic, and heterogeneous environments. It also established the diagnostic stage of the thesis, since the subsequent technical contributions are positioned as responses to some of the limitations and opportunities identified in this foundational analysis. These considerations led to the first publication of this PhD thesis, [Decentralized Federated Learning: Fundamentals, State of the Art, Frameworks, Trends, and Challenges \(Article 1–IEEE_COMST\)](#).

After completing the state-of-the-art analysis and identifying the main gaps in DFL, the methodology then focused on the adversarial dimension of decentralized learning, addressing **RQ2** and objectives **O.3** and **O.4**. This transition was motivated by the need to make DFL more reliable and robust in realistic decentralized settings, where the absence of a central coordinator shifts reliability assessment, aggregation decisions, and responses to perturbations to the participating nodes. In this stage, the research investigated how DFL could be reinforced in dynamic scenarios in which participants collaborate without relying on a central coordinating entity, and where malicious or unreliable behavior may affect the learning process. Military aerial reconnaissance was considered a demanding application setting for evaluating this research direction, as it combines mobile participants, changing connectivity, predefined operational constraints, and potential perturbations affecting both the mission context and the collaborative learning process. Accordingly, the methodological design combined threat modeling, the development of a solution based on situational awareness, and controlled empirical evaluation to investigate whether contextual, operational, and model-related information could support reliability-oriented decentralized aggregation under adversarial conditions.

The evaluation of this second stage was conducted in a bounded military aerial reconnaissance scenario using Synthetic Aperture Radar (SAR) data, four aerial vehicles, and various flight formations under adversarial conditions that affected both the mission context and the collaborative training process. This setting enabled evaluation of whether contextual and model-related indicators could support decentralized aggregation decisions when participants are subject to mobility and communication constraints, operational perturbations, and potentially malicious behavior [21]. By combining these conditions in a controlled scenario, this stage examined how situational awareness can be used to reinforce robust and adaptive DFL under bounded adversarial assumptions. The resulting work led to the second publication of this PhD thesis, [Flighter: Decentralized Federated Learning and Situational Awareness for Secure Military Aerial Reconnaissance \(Article 2–IEEE_COMMAG\)](#).

Once the adversarial and operational dimension of DFL were studied, the third methodological stage focused on statistical and multimodal heterogeneity in decentralized federated environments, tackling **RQ3**, **RQ4**, and objectives **O.5** and **O.6**. This stage covered scenarios in which clients differ in their local data distributions, class distributions are non-IID, modalities may be missing, and exchanging complete model parameters or full models may create excessive communication overhead [30]. To address this problem, a prototype-based technical solution for multimodal DFL was designed. The methodology was based on exchanging compact, modality-aware class representations rather than complete model parameters or gradients, allowing participants to share knowledge with neighboring clients in a decentralized manner while reducing communication overhead [36]. In this stage,

robustness was characterized primarily by the ability to preserve collaborative learning performance under heterogeneous client data distributions, incomplete modality availability, and constrained communication resources. This methodological decision directly links heterogeneity management to communication efficiency, two dimensions that are central to the general objective of the thesis.

The methodological design of this third contribution incorporated mechanisms to handle missing modalities, align representations, weight modality contributions, and fuse multimodal information under heterogeneous conditions. It also included theoretical analysis and experimental validation. The theoretical component examined convergence and communication costs, while the empirical component evaluated the approach on established multimodal benchmarks, non-IID data configurations, missing-modality settings, and comparisons with relevant federated and decentralized baselines [29]. In this way, the third methodological stage combined model design, decentralized knowledge exchange, formal analysis, and empirical assessment without assuming the presence of a central aggregation entity. This stage resulted in the third publication of this PhD thesis, [Decentralized Federated Learning with Multimodal Prototypes for Heterogeneous Data \(Article 3–ELSEVIER_IF\)](#).

In general, these methodological stages form a coherent storyline that links the foundational characterization of DFL with two targeted technical dimensions: adversarial reliability through situational awareness and heterogeneous multimodal learning through communication-efficient prototype exchange. This progression provides the methodological basis for the next section, which synthesizes the analytical, empirical, and theoretical results obtained from the three publications.

4 Results

The results of this PhD thesis are synthesized according to the objectives presented in Section 2. Since the thesis is conceived as a compendium of publications with an integrative memory, the results do not derive from a single experimental campaign. Instead, they emerge from a coherent sequence of three scientific contributions addressing DFL from complementary perspectives. The first contribution provides the foundational and analytical basis of the thesis by offering a systematic characterization of DFL. It organizes the field around architectural, communication, security, evaluation, and application dimensions, and identifies trends, lessons learned, and open challenges. The second contribution moves from conceptual analysis to technical evaluation by studying DFL in dynamic and adversarial scenarios using a solution based on situational awareness. The third contribution addresses heterogeneous and multimodal DFL through a prototype-based solution designed to operate under non-IID data distributions, missing modalities, and communication constraints. Together, these results support the general objective of advancing robust, adaptive, and communication-efficient DFL in heterogeneous environments.

The publication [Article 1–IEEE_COMST](#) provides a comprehensive study of DFL and the conceptual foundation needed to understand the remainder of the dissertation. In the reviewed literature, the publication also positioned DFL as requiring a dedicated analysis rather than treating it only as a subsidiary topic within broader surveys on CFL. The work identified and organized the main fundamentals that define DFL as a research paradigm, including federation architectures, network topologies, communication mechanisms, security and privacy approaches, key performance indicators, and optimization mechanisms. It also reviewed available frameworks, application scenarios, current trends, lessons learned, and future challenges. This contribution is relevant because it establishes that DFL cannot be understood only as the removal of the central server from classical FL.

Instead, decentralization redefines the distribution of responsibilities among participants and introduces new challenges in coordination, communication, trust distribution, robustness, heterogeneity, and performance assessment.

The survey identified that DFL mitigates some limitations of CFL, particularly dependence on a central aggregation entity, the presence of a single point of failure, and bottlenecks associated with server-based coordination. However, it also introduces specific technical challenges. These include the need to manage decentralized aggregation, maintain communication among participants, handle heterogeneous data and devices, define suitable topologies, assess performance through appropriate KPIs, and address security and privacy concerns without relying on a central trusted authority. A central result of [Article 1–IEEE_COMST](#) is therefore the identification of DFL as a paradigm whose practical adoption requires mechanisms specifically designed for decentralized and heterogeneous environments, rather than direct adaptations of certain CFL assumptions.

This first contribution supports **O.1** and **O.2**, which together provide the basis for answering **RQ1**. Regarding **O.1**, it offers a structured analysis of the state of the art in DFL and defines the conceptual and technical elements used throughout the thesis. With respect to **O.2**, it identifies limitations, trends, and open challenges that motivate subsequent technical contributions. In particular, the survey explains why robustness, adaptation, communication efficiency, and heterogeneity management are central issues in DFL.

The results of [Article 1–IEEE_COMST](#) further contribute to the global storyline of the thesis by identifying the main application scenarios where DFL may be relevant. These scenarios include healthcare, Industry 4.0, mobile services, military environments, and vehicular systems, among others. The relevance of these scenarios lies not only in their diversity, but also in the recurring constraints that they expose: data are distributed, centralization may be undesirable or impractical, participants may be heterogeneous, and communication conditions may vary over time. In the context of this thesis, this scenario-level analysis serves as a bridge between the foundational survey and the subsequent technical contributions. It shows that the challenges identified in [Article 1–IEEE_COMST](#) can be instantiated along two complementary research directions: decentralized reliability under dynamic and adversarial conditions, addressed in [Article 2–IEEE_COMMAG](#), and communication-efficient learning under heterogeneous multimodal data availability, addressed in [Article 3–ELSEVIER_IF](#). For this reason, the adversarial aerial reconnaissance scenario considered in [Article 2–IEEE_COMMAG](#) and the heterogeneous multimodal learning scenario considered in [Article 3–ELSEVIER_IF](#) are not treated as isolated case studies, but as targeted responses to two major challenges identified in the foundational analysis.

The publication [Article 2–IEEE_COMMAG](#) focuses on the robustness and adaptation of DFL in dynamic and adversarial scenarios. It proposes *Flighter*, a situational-awareness solution for DFL, and evaluates it in a bounded adversarial aerial reconnaissance scenario. The publication considers several aerial vehicles that collaborate to train models for enemy vehicle detection without relying on a central coordinating server. This setting is demanding because participants are mobile, communication links may be unstable, and adversaries may affect both the operational context and the collaborative learning process [20]. The main result of this publication is that situational awareness can be integrated into DFL to support decentralized reliability decisions during collaborative training.

Flighter incorporates contextual, operational, and model-related information into the defense process. Its defense module combines model similarity with situational-awareness indicators: flight formation, geopositioning, interconnection behavior, and resource usage. These indicators reflect limitations previously identified for DFL in dynamic and adversarial settings, where participants may be mobile, communication links may be unstable, opera-

tional behavior may deviate from expected patterns, and malicious contributions may affect decentralized aggregation. The indicators are normalized and combined into a weighted defense score that represents the estimated reliability of each participant. This score is evaluated against a dynamic threshold that adapts during training. Participants whose scores exceed the threshold receive greater influence during decentralized aggregation, whereas the aggregation weights of participants below the threshold are reduced. If low reliability persists across multiple rounds, the corresponding connections may be severed to protect the collaborative learning process. This design addresses **O.3** and contributes to answering **RQ2** by showing how a situational-awareness solution can support reliability-oriented decentralized aggregation in a dynamic and adversarial environment. Consequently, *Flighter* is considered in this thesis to be a bounded contribution to robustness and adaptation in adversarial DFL, rather than a general-purpose defense mechanism for every decentralized federated setting.

The empirical results of [Article 2–IEEE_COMMAG](#) support **O.4**, further contributing to **RQ2**. The publication evaluated *Flighter* in a simulated military aerial reconnaissance scenario involving four SAR-equipped aircraft, several flight formations, and adversarial conditions affecting geopositioning, trajectory behavior, and collaborative training. The quantitative results reported in the paper correspond to the VGG16-based evaluation on three SAR datasets: *MSTAR* for military ground targets [37], *SAMPLE* for measured and synthetic SAR target pairs [38], and *OpenSARShip* for maritime SAR ship classification [39]. The comparison was performed against a state-of-the-art DFL baseline that mirrored the training process, model architecture, data processing, and federation rounds, but excluded the defense module based on situational awareness. In the baseline reconnaissance scenario, *Flighter* achieved F1-score values of 95.8% on *MSTAR*, 97.5% on *SAMPLE*, and 79.1% on *OpenSARShip*. Resource usage remained moderate in this configuration, with CPU usage around 57% and RAM usage around 1.7 GB, while packet loss and round times remained stable, around 0.7% packet loss and 5.3 s per round in the baseline setting. Despite geopositioning deviations, *Flighter* maintained F1-score values of 91.8% on *MSTAR*, 90.3% on *SAMPLE*, and 72.0% on *OpenSARShip*, with packet loss below 1% and only slight increases in round time. The evaluation also reported F1-score values of 92.5%, 89.8%, and 70.1% under collision-course manipulation on *MSTAR*, *SAMPLE*, and *OpenSARShip*, respectively. In contrast, adversarial data-poisoning scenarios yielded F1-score values ranging from 52.1% to 90.1%, depending on the dataset and attack configuration. These results should be interpreted in the context of the specific datasets, attacks, model architecture, baseline, and simulation conditions reported in the publication.

The publication [Article 3–ELSEVIER_IF](#) addresses heterogeneous and multimodal DFL. It examines collaborative learning scenarios in which participants present non-IID data distributions, uneven availability of modalities, missing modalities, and communication constraints. To address these challenges, the publication proposes *Modalis*, a prototype-based solution for multimodal DFL built around a communication-efficient prototype-exchange protocol. Rather than transmitting complete model parameters or high-dimensional updates, *Modalis* enables participants to exchange compact, modality-aware class prototypes that summarize class-discriminative information learned from heterogeneous local data. This design is central to the thesis because it connects two major challenges identified in the survey: heterogeneity management and communication efficiency. In addition to empirical validation, the publication provides theoretical support through a convergence result and a communication-cost analysis, framing prototype exchange as a mechanism that contributes to both multimodal collaborative learning and communication-efficient decentralized collaboration.

Beyond the communication protocol, Modalis incorporates mechanisms for handling missing modalities, aligning multimodal representations, weighting modality contributions, and fusing information under heterogeneous data availability. In particular, contextual null embeddings provide substitute representations when a modality is absent, adaptive gating adjusts the relevance of available modalities, and the multimodal fusion component integrates partial information into a shared representation. These mechanisms are complemented by representation-alignment objectives that encourage consistency between local multimodal embeddings and exchanged prototypes. As a result, participants can benefit from decentralized collaboration even when their local data are incomplete, statistically skewed, or limited to a subset of the modalities available in the federation. This result supports **O.5** and contributes to answering **RQ3** by showing that a prototype-based approach can be designed and evaluated for DFL with non-IID class distributions and missing modalities across participants, without relying on a central entity to perform multimodal fusion or handle missing modalities.

The experimental results reported in [Article 3–ELSEVIER_IF](#) show improved performance under heterogeneous multimodal DFL conditions. The evaluation considered four multimodal datasets: AVMNIST for audio-visual spoken digit recognition [40], CREMA-D for audio-visual emotional speech analysis [41], UCI-HAR for human activity recognition using accelerometer and gyroscope data [42], and CRISIS-MMD for image-text crisis event classification from social media [43]. Non-IID class skew was controlled through $\beta \in 1.0, 0.5, 0.1$, and missing-modality configurations were controlled through $p_m \in 0.0, 0.25, 0.5$. The experiments involved 20 clients connected through a static Erdős–Rényi graph with $p_{ER} = 0.9$ for the main reported results, and Modalis was compared against a centralized upper bound and several FL/DFL baselines, including FedAvg, FedProto, FedInMM, CreamFL, PmcmFL, FedMAC, and MFCPL [31, 28, 29, 34]. Under high heterogeneity, the article reports a relative F1-score gain of up to approximately 4.0% over the next-best competing method. The results also show that performance degradation increases as label skew and missing information become more severe, although Modalis generally exhibits smaller relative performance losses than the compared methods.

A particularly relevant result concerns clients with only one available modality. In the most challenging configuration analyzed in the article, Modalis supported cross-modal knowledge transfer for unimodal clients, achieving an F1 score of 82.1% on AVMNIST for image-only clients, corresponding to a 19.9% relative gain over FedAvg. This result suggests that prototype exchange can help decentralized participants benefit from knowledge associated with modalities not locally available, although this interpretation remains contingent on the datasets, modality configurations, baselines, and experimental assumptions documented in the publication. The article also reports a cross-dataset transferability analysis in which an AVMNIST-tuned Modalis configuration incurred only modest F1-score decreases on CREMA-D, UCI-HAR, and CRISIS-MMD while retaining the best absolute scores among the compared methods. These results support robustness to hyperparameter transfer under the evaluated conditions, but they do not imply zero-shot generalization across incompatible datasets.

Modalis also supports **O.6** and contributes to answering **RQ4**, through its communication-efficiency results. In the computation-intensive AVMNIST configuration with high label skew and missing modalities, Modalis reported an average communication cost of approximately 0.11 MB per client message and an F1 score of 83.4%. This communication payload is more than 40× smaller than that of communication-heavy methods such as FedAvg and PmcmFL, which transmit around 4.75–4.80 MB per round. This result is central to the thesis because communication efficiency is a structural requirement for DFL: in decentralized

settings, excessive exchange of model parameters may limit scalability, increase latency, or reduce applicability in resource-constrained scenarios. The resource analysis also reported approximately 44% CPU utilization, 70% GPU utilization, and 31% RAM utilization on the evaluation node, placing Modalis close to advanced multimodal FL baselines in terms of computational footprint.

Taken together, the three publications provide the analytical and empirical basis for addressing the doctoral problem from complementary perspectives. [Article 1–IEEE_COMST](#) establishes the conceptual structure of DFL and identifies the challenges that motivate the subsequent technical work. [Article 2–IEEE_COMMAG](#) demonstrates, in a bounded adversarial aerial reconnaissance scenario, how situational awareness can support reliability-oriented decentralized aggregation amid mobility, communication instability, and adversarial behavior. [Article 3–ELSEVIER_IF](#) shows how multimodal prototype exchange can support collaborative learning under non-IID data, missing modalities, and communication constraints. These results connect the foundational characterization of DFL with two targeted technical responses to robustness, heterogeneity, and communication-efficiency challenges in decentralized federated environments.

5 Conclusions and future work

This PhD thesis has studied how to advance DFL toward more robust collaborative learning in heterogeneous, adversarial, and communication-constrained environments through three core journal publications that address four complementary research questions. **RQ1** examined the fundamentals, frameworks, application scenarios, metrics, limitations, and challenges that define DFL. **RQ2** focused on how reliability and robustness can be reinforced against adversarial behavior, operational perturbations, and potentially compromised nodes. **RQ3** addressed decentralized multimodal learning under non-IID data and missing-modality conditions, while **RQ4** examined the effect of efficient information-exchange mechanisms on communication cost and collaborative learning performance. These questions were addressed through three complementary contributions: a foundational characterization of DFL [Article 1–IEEE_COMST](#); a solution based on situational awareness evaluated in a bounded adversarial scenario [Article 2–IEEE_COMMAG](#); and a prototype-based solution for heterogeneous multimodal DFL [Article 3–ELSEVIER_IF](#). Together, these contributions support the central argument of the thesis: DFL can reduce some structural limitations of CFL, but its practical adoption requires mechanisms specifically designed for decentralized, heterogeneous, dynamic, and potentially adversarial environments.

With respect to **RQ1**, the thesis concludes that DFL constitutes a specific form of collaborative learning that reduces the need to share raw data, in which the absence of a central aggregation entity changes how coordination, communication, aggregation, trust, and evaluation must be designed. The analysis conducted in [Article 1–IEEE_COMST](#) shows that decentralization redistributes responsibilities across the federation, including communication, aggregation, topology management, robustness, privacy, trust distribution, and performance assessment. This contribution addresses **O.1** and **O.2** by organizing the main foundations of DFL and identifying the limitations, trends, lessons learned, and open challenges that motivate the development of robust, adaptive, and communication-efficient mechanisms for heterogeneous and dynamic environments.

For **RQ2**, the thesis concludes that, within the bounded adversarial conditions evaluated, combining model-related information with contextual and operational awareness can support reliability-oriented decentralized aggregation and improve robustness against the considered perturbations. The contribution developed in [Article 2–IEEE_COMMAG](#) proposed Flighter

as a technical solution for dynamic and adversarial DFL scenarios. By incorporating indicators such as model similarity, flight formation, geopositioning, resource usage, and interconnection rate, Flighter supports reliability-oriented aggregation decisions without relying on a central supervisor. Its evaluation in a bounded SAR-based aerial reconnaissance scenario indicates that situational awareness can support decentralized robustness under the adversarial conditions considered. This contribution addresses **O.3** and **O.4**, while keeping the military setting as an evaluation scenario rather than the general thematic scope of the thesis.

Regarding **RQ3** and **RQ4**, the thesis concludes that prototype-based knowledge exchange is a suitable mechanism for supporting decentralized multimodal learning under non-IID data and missing-modality conditions, while reducing communication costs in the evaluated settings. The contribution reported in [Article 3–ELSEVIER_IF](#) proposed Modalis, a prototype-based technical solution that replaces full model-parameter exchange with compact, modality-aware class prototypes. These representations allow participants to share discriminative knowledge with neighboring clients while supporting missing-modality handling, representation alignment, modality weighting, and multimodal information fusion. The reported results indicate that Modalis improves collaborative learning in the heterogeneous configurations evaluated and reduces communication costs compared to full-model exchange approaches. This contribution addresses **O.5** and **O.6** by linking the management of multimodal heterogeneity to communication-efficient decentralized collaboration.

The scope of the thesis is also defined by several limitations. First, the foundational analysis conducted in [Article 1–IEEE_COMST](#) is necessarily time-bound, since DFL is an active research area in which new frameworks, optimization strategies, privacy mechanisms, and application scenarios continue to emerge. Moreover, although the work examined security and privacy as fundamental dimensions of DFL, it did not provide a dedicated threat-centric analysis of adversaries, attack surfaces, and attack–defense relationships, which have subsequently received more specialized attention [8]. Its taxonomy, trends, lessons learned, and open challenges should therefore be understood as a structured characterization of the field at the time of publication, rather than as a definitive or exhaustive description of DFL. Second, the two technical contributions, [Article 2–IEEE_COMMAG](#) and [Article 3–ELSEVIER_IF](#), were evaluated in controlled evaluation settings, which support analytical clarity and reproducibility but limit direct generalization to operational deployments. In the case of Flighter, the main limitations concern external validity beyond the simulated setting, scalability to larger and more heterogeneous federations, and coverage of additional adversarial strategies, communication regimes, and operational disturbances. In the case of Modalis, the main limitations concern benchmark-driven evaluation, controlled heterogeneity generation, and predefined assumptions about decentralized communication. Further validation would be needed to assess its behavior under naturally heterogeneous data streams, dynamic topologies, heterogeneous device capabilities, energy constraints, and real edge-network conditions. Finally, although prototype exchange reduces communication costs by avoiding transmission of full model parameters, compact representations should not be interpreted as providing formal privacy protection on their own. A dedicated privacy analysis, which could include differential privacy, secure aggregation, or leakage assessment mechanisms, remains outside the empirical scope of this thesis. These limitations delimit the methodological and empirical conditions under which the conclusions should be interpreted.

The results of this thesis open up several lines for future research. A first direction is to evaluate DFL solutions under deployment conditions that are closer to those expected in real decentralized systems. The evaluations conducted in this thesis provide controlled evidence on specific adversarial and heterogeneous settings, but real deployments may involve larger

numbers of participants, dynamic neighborhood relationships, intermittent availability, heterogeneous computational capabilities, and unstable communication links. These factors should not be treated as isolated sources of degradation, as changes in topology, participant availability, communication quality, and device resources may jointly affect convergence, robustness, and communication efficiency. Future work should therefore examine how DFL mechanisms behave when these constraints evolve simultaneously during training, especially in edge-oriented and mobile collaborative environments where the federation cannot be assumed to remain stable over time.

The second direction concerns extending situational awareness as a reliability mechanism for decentralized collaboration. In *Flighter*, contextual and operational indicators are used to support reliability-oriented aggregation decisions under bounded adversarial conditions. A relevant next step is to study how such indicators should be selected, weighted, updated, and validated when the operating context changes or when the available contextual information is incomplete, noisy, or potentially manipulated. Addressing this research direction requires moving beyond the specific indicators and assumptions evaluated in the thesis to examine whether the reliability scores remain meaningful when participants differ in mobility patterns, communication quality, computational resources, and behavioral consistency. Future research should also analyze how much contextual information is needed to improve robustness without introducing excessive communication overhead or new attack surfaces.

A third line of future work is the formal privacy analysis of prototype-based DFL. *Modalis* reduces communication costs by exchanging compact, modality-aware class prototypes instead of full model parameters, but these compact representations should not be assumed to provide privacy guarantees on their own. Prototypes may still encode information about local class distributions, feature representations, or patterns of modality availability, especially when clients have small or highly skewed datasets. Future research should quantify the potential leakage associated with prototype exchange and study privacy-preserving mechanisms specifically adapted to decentralized prototype learning. Such research may include differential privacy mechanisms, privacy-aware prototype construction, secure aggregation among neighboring nodes, or empirical leakage assessment methods that evaluate what an adversary could infer from the exchanged representations.

In summary, this PhD thesis advances the study of robust DFL in heterogeneous environments through a coherent sequence of analytical and technical contributions. The thesis shows that DFL can reduce dependence on a central aggregation entity while also demonstrating that its practical adoption requires mechanisms designed for the challenges introduced by decentralization. By characterizing the field, designing and evaluating a situational-awareness-based technical solution for bounded adversarial scenarios, and developing a prototype-based technical solution for heterogeneous multimodal learning, the thesis contributes to a precise and empirically grounded understanding of how DFL can evolve toward robust, adaptive, and communication-efficient collaborative learning.

1 Introducción y motivación

La creciente distribución de los datos, la computación y la toma de decisiones a través de las infraestructuras digitales modernas han transformado el diseño, el entrenamiento y el despliegue de modelos de Aprendizaje Automático (ML). Los entornos conectados actuales ya no están compuestos únicamente por sistemas aislados ni por repositorios centralizados de datos. Cada vez implican a más participantes, organizaciones, dispositivos, sensores y nodos frontera (*edge*) que generan y procesan información localmente. Esta evolución es particularmente relevante en el Internet de las Cosas (IoT) y en otros entornos distribuidos, donde los datos se generan, se procesan y están gobernados por múltiples participantes en contextos industriales, sanitarios, vehiculares, tácticos y ciberfísicos, entre otros. En estos escenarios, los datos están distribuidos de forma natural y centralizarlos puede resultar impracticable, costoso o indeseable debido a restricciones técnicas, regulatorias, de privacidad u operativas [1, 2]. En este contexto, la capacidad de extraer conocimiento a partir de fuentes de información distribuidas es esencial para apoyar la monitorización, la automatización y la toma de decisiones basadas en datos. Sin embargo, la diversidad de participantes y el dinamismo del entorno aumentan la complejidad de construir sistemas de aprendizaje fiables, eficientes y compatibles con la propiedad local de los datos. Estas restricciones motivan mecanismos de aprendizaje que operen cerca del proceso que genera los datos y que, al mismo tiempo, permitan a los participantes beneficiarse del entrenamiento colaborativo de modelos.

La Inteligencia Artificial (IA), y particularmente ML, se ha convertido en un mecanismo fundamental para explotar el valor de los datos distribuidos. No obstante, los enfoques convencionales de ML suelen asumir que los datos pueden recopilarse, transferirse y procesarse en una infraestructura centralizada. Esta suposición resulta cada vez más difícil de mantener cuando los participantes están geográficamente distribuidos, los datos pueden ser sensibles, los dispositivos operan con recursos computacionales o energéticos limitados, y los enlaces de comunicación son inestables o intermitentes [3]. En tales casos, transferir todos los datos locales a una ubicación central puede implicar restricciones técnicas y organizativas. Puede aumentar los costes de comunicación y la latencia, exponer información sensible y crear dependencias de gobernanza u operativas respecto a una única infraestructura [4, 5]. Estas limitaciones son más pronunciadas cuando los datos se generan de forma continua en el *edge* o cuando los participantes pertenecen a distintos dominios administrativos. Por consiguiente, el problema pasa de entrenar modelos precisos de forma aislada a diseñar

mecanismos de aprendizaje colaborativo que exploten la información local sin asumir que todos los datos pueden agregarse en un único punto para su procesamiento.

El Aprendizaje Federado (FL), propuesto originalmente por Google en 2016, surgió como un paradigma de aprendizaje colaborativo diseñado para apoyar el entrenamiento de modelos con preservación de la privacidad sin necesidad de almacenar datos sin procesar en un único punto central [6]. En FL, los participantes entrenan modelos con sus datos locales e intercambian información sobre los modelos en lugar de conjuntos de datos brutos. Este enfoque permite desarrollar capacidades de aprendizaje compartido manteniendo los datos almacenados localmente, lo cual es particularmente relevante para el ML distribuido y consciente de la privacidad. La configuración más común de FL ha seguido tradicionalmente una arquitectura centralizada, en la que un servidor coordina el proceso de entrenamiento, recibe actualizaciones locales de los modelos, las agrega y redistribuye el modelo global resultante [4]. En consecuencia, el Aprendizaje Federado Centralizado (CFL) se ha convertido en un enfoque ampliamente estudiado, con un extenso trabajo sobre optimización, comunicación, privacidad, personalización, robustez y diseño de sistemas en múltiples dominios de aplicación [5]. No obstante, la efectividad de CFL depende de que la entidad coordinadora sea capaz de gestionar la comunicación, la agregación, la sincronización y la cadena de confianza a lo largo de todo el proceso de entrenamiento.

Aunque sus ventajas lo hacen adecuado en determinadas circunstancias, CFL introduce limitaciones estructurales que resultan críticas en varios escenarios reales. El servidor central puede convertirse en un cuello de botella a medida que aumenta el número de participantes o el tamaño de los modelos intercambiados. También puede convertirse en un único punto de fallo cuando la entidad coordinadora falla, se sobrecarga, es atacada o se desconecta [7, 8]. Además, la coordinación centralizada concentra la confianza en una sola entidad, lo que puede resultar inadecuado para entornos compuestos por participantes autónomos, nodos móviles, diferentes dominios administrativos o infraestructuras en las que no puede asumirse un coordinador estable y universalmente confiable [9]. Estas limitaciones son particularmente evidentes en entornos distribuidos, móviles y tácticos, donde los enlaces de comunicación cambian con el tiempo y los participantes pueden necesitar operar con conectividad parcial o con restricciones operativas. Bajo estas condiciones, la arquitectura basada en un único servidor centralizado puede interrumpir la continuidad del proceso de aprendizaje y reducir la capacidad del sistema para adaptarse a cambios en la topología, los recursos o la disponibilidad. Estas limitaciones justifican la investigación sobre arquitecturas alternativas de aprendizaje colaborativo en las que las responsabilidades de agregación y coordinación no estén concentradas en una única entidad.

El Aprendizaje Federado Descentralizado (DFL) surge en este contexto como un paradigma en el que los participantes colaboran sin depender de un servidor central de agregación. En lugar de enviar actualizaciones a un coordinador único, los participantes intercambian información directamente entre sí mediante comunicación entre pares, agregación basada en vecinos, mecanismos de consenso o estrategias de coordinación descentralizada [7, 10]. Esta descentralización puede reducir la dependencia de una infraestructura central determinada y mejorar la tolerancia a fallos, la escalabilidad y la resiliencia en escenarios con conectividad intermitente o en los que no se puede garantizar la existencia de un servidor central de confianza [8, 9]. Por tanto, DFL es especialmente valioso cuando los participantes autónomos necesitan colaborar en condiciones distribuidas, con restricciones de recursos y conectividad intermitente [1]. Sin embargo, la descentralización también modifica las suposiciones del proceso de aprendizaje. Los participantes pueden tener solo conocimiento parcial de la federación, comunicarse con un conjunto limitado de vecinos y tomar decisiones de agregación a partir de la información disponible localmente. En

consecuencia, DFL no es solo una variación arquitectónica de CFL, sino un paradigma distinto de aprendizaje colaborativo con sus propios retos en los ámbitos del sistema, la comunicación, la optimización y la seguridad.

La ausencia de un coordinador central cambia la naturaleza del problema de aprendizaje. En CFL, responsabilidades como la agregación, la sincronización, la coordinación de participantes, la evaluación de la fiabilidad y, en algunos casos, la monitorización de la seguridad suelen ser gestionadas por el servidor central. En DFL, estas responsabilidades deben distribuirse, en cambio, entre los nodos participantes [10]. Esta redistribución introduce nuevos retos relacionados con la gestión de la comunicación, el diseño de topologías, la convergencia, la robustez, la distribución de la confianza, la privacidad, la seguridad y la evaluación del rendimiento [11]. El diseño de sistemas DFL requiere una consideración cuidadosa de aspectos como la coordinación, la comunicación, la agregación, la evaluación de la fiabilidad, los criterios de evaluación y los tipos de información intercambiada entre los participantes. Además, la propia estructura de comunicación se convierte en parte del mecanismo de aprendizaje, ya que la topología determina cómo se propaga el conocimiento a través de la federación y cómo los sesgos locales pueden afectar a los participantes vecinos. Por esta razón, el valor de DFL no reside meramente en eliminar un servidor central, sino en proporcionar mecanismos que permitan a participantes descentralizados colaborar de forma fiable bajo condiciones realistas.

Una primera implicación de esta redistribución concierne a la fiabilidad y la robustez. Dado que los participantes de DFL toman decisiones de comunicación y agregación basadas en información localmente disponible, el sistema debe preservar el aprendizaje colaborativo incluso cuando algunos nodos no son fiables, las condiciones de comunicación varían o las actualizaciones intercambiadas pueden verse afectadas por un comportamiento adversarial. Los participantes pueden verse afectados por fallos de comunicación, movilidad, perturbaciones operativas o comportamiento comprometido [12]. En arquitecturas centralizadas, un servidor puede apoyar tareas de monitorización, filtrado o coordinación. En DFL, sin embargo, no existe una autoridad central con una visión global de la federación, y la superficie de ataque se redistribuye mediante los procesos de comunicación y agregación [13, 14]. Esta falta de supervisión centralizada incrementa la necesidad de mecanismos descentralizados capaces de estimar la fiabilidad de la información intercambiada, de reducir la influencia de contribuciones potencialmente dañinas y de preservar el aprendizaje colaborativo en condiciones adversas [15, 16]. Por tanto, la robustez en DFL depende no solo del algoritmo de aprendizaje, sino también de la interacción entre la topología, la comunicación, el comportamiento de los participantes y los criterios utilizados para aceptar, ponderar o descartar la información. La naturaleza distribuida de la federación amplifica estos problemas, ya que un participante comprometido o poco fiable puede influir en sus vecinos y, posteriormente, afectar a otras partes de la red.

En entornos DFL dinámicos y adversariales, la robustez no puede evaluarse únicamente por la consistencia de las actualizaciones del modelo. La fiabilidad de una contribución también puede depender del contexto operacional en el que se genera, incluyendo la estabilidad de la comunicación, la disponibilidad de recursos, la movilidad, las condiciones de sensorización y el comportamiento potencialmente comprometido. Estos factores son particularmente relevantes porque los enfoques actuales de DFL siguen ofreciendo un apoyo limitado para escenarios en los que la movilidad, las comunicaciones no fiables, el comportamiento adversarial y la agregación descentralizada deben gestionarse conjuntamente [17]. La conciencia situacional, entendida tradicionalmente como la percepción, comprensión y proyección de elementos relevantes en un entorno cambiante, proporciona una base conceptual para conectar la evidencia a nivel de modelo con la información contextual en la

colaboración descentralizada [18]. Desde esta perspectiva, un DFL robusto requiere mecanismos conscientes de la fiabilidad que evalúen la información intercambiada no solo por su comportamiento en los procesos de aprendizaje, sino también por las condiciones en las que los participantes interactúan. Los escenarios tácticos y orientados a la defensa constituyen un ejemplo claro de estas condiciones, ya que combinan la colaboración descentralizada con la movilidad, las limitaciones de comunicación, la incertidumbre operativa y un comportamiento potencialmente adversarial [19]. Dentro de esta categoría, el reconocimiento aéreo militar resulta particularmente relevante, puesto que los UAV y otros vehículos aéreos militares pueden necesitar realizar conjuntamente tareas de sensorización, computación, comunicación y aprendizaje mientras cambian de posición, operan con conectividad intermitente y se enfrentan a acciones hostiles que afectan tanto a las condiciones de la misión como al propio proceso de entrenamiento colaborativo [20, 21]. Por tanto, este ámbito proporciona un escenario acotado pero exigente en el que la fiabilidad del DFL puede examinarse tanto desde la perspectiva del modelo como desde la operativa.

Un segundo reto importante concierne a la heterogeneidad, que afecta tanto a los nodos participantes como a los datos que poseen. Los entornos descentralizados reales rara vez están compuestos por participantes con datos, recursos, conectividad o condiciones operativas idénticos. Los dispositivos pueden diferir en recursos, conectividad, movilidad, disponibilidad de datos locales y patrones de participación a lo largo del tiempo [22]. Sus conjuntos de datos también pueden seguir distribuciones que no sean independientes entre sí ni idénticas (non-IID), lo que significa que los participantes pueden observar diferentes clases, distribuciones de características, proporciones de etiquetas o patrones locales de datos [11, 23]. Estas formas de heterogeneidad pueden afectar la estabilidad del entrenamiento, la calidad de la agregación, el comportamiento de los procesos de convergencia y el rendimiento final del modelo [24, 22]. En CFL, un servidor central puede coordinar la participación, aplicar reglas globales de agregación o monitorizar el comportamiento de las actualizaciones locales. En DFL, estas responsabilidades están distribuidas y el efecto de la heterogeneidad puede propagarse a través de vecindarios locales. Por esta razón, DFL requiere mecanismos que puedan operar bajo condiciones desiguales de recursos, distribuciones de datos sesgadas y una visibilidad parcial de la federación global.

La heterogeneidad se vuelve aún más compleja cuando los escenarios de aprendizaje involucran múltiples modalidades de datos. Muchas aplicaciones no dependen de una única fuente de información, sino que combinan imágenes, audio, texto, mediciones de sensores, telemetría, señales estructuradas u otras formas de datos [25, 26]. En entornos descentralizados, no todos los participantes necesariamente tienen acceso a las mismas modalidades. Algunos clientes pueden observar solo una modalidad, otros pueden tener muestras multimodales completas y otros pueden experimentar modalidades ausentes en parte de sus datos locales [27]. Esta situación plantea un reto significativo, ya que el conocimiento debe intercambiarse y fusionarse sin asumir una disponibilidad homogénea de datos entre los participantes. Cuando las modalidades están ausentes o distribuidas de manera desigual, los mecanismos convencionales basados en el intercambio de parámetros completos del modelo pueden volverse ineficientes o insuficientemente adaptados a la estructura del problema [28, 29]. Además, el aprendizaje multimodal requiere no solo la agregación, sino también la alineación de representaciones, la fusión de modalidades y la capacidad de preservar información discriminativa entre clientes con observaciones locales distintas. Por esta razón, el DFL multimodal requiere estrategias de comunicación y aprendizaje capaces de representar información útil incluso cuando los datos locales son incompletos, están estadísticamente desbalanceados y están distribuidos entre participantes con diferentes capacidades de sensorización [30]. Este requisito motiva mecanismos de

comunicación a nivel de representación que no dependan exclusivamente del intercambio de parámetros completos del modelo o gradientes.

El aprendizaje basado en prototipos ha ganado atención en este contexto porque las representaciones compactas, a nivel de clase o conscientes de la modalidad, pueden facilitar el intercambio de conocimiento al tiempo que reducen la carga de comunicación asociada a las actualizaciones de modelos de alta dimensionalidad. En CFL, el intercambio de prototipos se ha utilizado para mitigar la heterogeneidad de datos mediante el intercambio de representaciones abstractas de clases en lugar de gradientes o de parámetros completos [31, 32]. En entornos descentralizados, también se ha explorado el intercambio de prototipos para reducir la sobrecarga de comunicación y mejorar la consistencia del modelo entre clientes vecinos [33]. Para FL multimodal, se han propuesto mecanismos de prototipos cross-modal para abordar condiciones severas de ausencia de modalidades y desalineación de representaciones [34], mientras que la ausencia parcial de modalidades ha motivado estrategias específicas de agregación y regularización en sistemas federados [35]. Estos desarrollos permiten la integración de datos heterogéneos, el intercambio de conocimiento a nivel de representación y la colaboración eficiente desde el punto de vista de las comunicaciones, sin requerir el intercambio sistemático de los parámetros completos del modelo. También ilustran una dirección de investigación más amplia en la que la información intercambiada durante la federación puede consistir en representaciones compactas que codifican conocimiento discriminante. Esta perspectiva es central para DFL, donde la comunicación ocurre entre los participantes, y reducir el consumo de red puede ser necesario para sostener la colaboración en condiciones de recursos limitados.

En general, la literatura sugiere que DFL es un paradigma relevante para el aprendizaje colaborativo en entornos distribuidos, pero su adopción práctica sigue dependiendo de un conjunto más amplio de retos fundacionales, metodológicos y operativos. Primero, el campo necesita una comprensión sistemática de sus fundamentos, incluyendo arquitecturas, topologías, mecanismos de comunicación, robustez, enfoques de seguridad y privacidad, indicadores clave de rendimiento, marcos de trabajo, escenarios de aplicación, tendencias y retos abiertos [7, 11]. Segundo, el aprendizaje descentralizado debe hacerse más robusto frente a condiciones adversariales y dinámicas, en las que no existe una autoridad central con una visión completa de la federación, y el comportamiento dañino puede afectar tanto a la comunicación como a la agregación [15]. Tercero, los sistemas DFL deben abordar la heterogeneidad estadística, de recursos, comunicacional y multimodal, particularmente cuando los clientes operan con distribuciones de datos non-IID, disponibilidad incompleta de modalidades o diferentes capacidades de comunicación [25]. Finalmente, se necesitan mecanismos de comunicación eficientes para hacer viable DFL en escenarios con recursos limitados, donde el intercambio completo de modelos puede resultar costoso o ineficiente.

Con base en las consideraciones anteriores, esta tesis doctoral investiga cómo DFL puede avanzar hacia un aprendizaje colaborativo robusto, adaptativo y eficiente en la comunicación en entornos heterogéneos. Estudia DFL como un paradigma que reduce la dependencia de una entidad central de agregación al tiempo que redistribuye las responsabilidades de comunicación, confianza, fiabilidad, robustez y evaluación del rendimiento entre los nodos participantes. Dentro del panorama de investigación más amplio de DFL, la tesis se centra en un subconjunto coherente de problemas: la caracterización fundacional del campo, la fiabilidad descentralizada bajo condiciones adversariales y dinámicas acotadas, y el aprendizaje eficiente en las comunicaciones bajo heterogeneidad estadística y multimodal, incluyendo distribuciones no IID y modalidades ausentes. Desde esta perspectiva, la tesis está guiada por las siguientes preguntas de investigación:

- **RQ1:** ¿Cuáles son los fundamentos, marcos de trabajo, escenarios de aplicación,

métricas, limitaciones y retos que definen el estado actual de DFL?

- **RQ2:** ¿Cómo pueden reforzarse la fiabilidad y la robustez de DFL frente al comportamiento adversarial, las perturbaciones operativas y los nodos potencialmente comprometidos?
- **RQ3:** ¿Qué mecanismos pueden apoyar el aprendizaje multimodal descentralizado bajo condiciones de datos non-IID y de modalidades ausentes, preservando al mismo tiempo el rendimiento del aprendizaje colaborativo?
- **RQ4:** ¿Cómo afectan los mecanismos de comunicación eficientes al coste de dicha comunicación y al rendimiento del aprendizaje colaborativo en DFL en escenarios multimodales heterogéneos?

2 Objetivos

El objetivo principal de esta tesis doctoral es avanzar DFL como paradigma para el aprendizaje colaborativo robusto, adaptativo y eficiente desde el punto de vista de las comunicaciones en entornos heterogéneos. Más específicamente, la tesis examina DFL como un enfoque de aprendizaje descentralizado en el que los participantes colaboran sin un agregador central, mientras se enfrentan a datos heterogéneos, condiciones de comunicación cambiantes y comportamientos potencialmente adversariales.

La tesis aborda tres aspectos complementarios de DFL, organizados en una progresión conceptual que va desde la caracterización fundacional hasta las respuestas técnicas específicas. En primer lugar, analiza los fundamentos de DFL, incluyendo arquitecturas, topologías, mecanismos de comunicación, enfoques de seguridad y privacidad, indicadores de rendimiento, marcos de trabajo, escenarios de aplicación, tendencias y retos abiertos. Este análisis fundacional identifica los retos planteados por la descentralización y subraya la necesidad de mecanismos que apoyen la fiabilidad, la robustez, la gestión de la heterogeneidad y la eficiencia de las comunicaciones. Sobre esta base, la tesis articula dos líneas técnicas complementarias derivadas de los retos identificados en el estado del arte. La primera línea examina la dimensión adversarial de DFL mediante el estudio de cómo la conciencia situacional puede apoyar la fiabilidad descentralizada en escenarios dinámicos afectados por perturbaciones operativas, ataques de comunicación y ataques al proceso de aprendizaje colaborativo. La segunda línea aborda la heterogeneidad estadística y multimodal mediante mecanismos de comunicación basados en prototipos, con especial atención a distribuciones de clases non-IID, modalidades ausentes y restricciones de eficiencia en la comunicación. Los siguientes objetivos específicos definen el alcance científico de la tesis e indican sus preguntas de investigación correspondientes:

- O.1.** Analizar el estado del arte actual relativo a DFL mediante una revisión, análisis y comparación exhaustiva de sus fundamentos, arquitecturas, topologías, mecanismos de comunicación, enfoques de seguridad y privacidad, indicadores clave de rendimiento, mecanismos de optimización, marcos de trabajo y escenarios de aplicación (**RQ1**).
- O.2.** A partir de la caracterización establecida en **O.1**, identificar y evaluar críticamente las principales limitaciones, brechas de investigación, tendencias y retos abiertos en DFL que evidencian la necesidad de mecanismos robustos, adaptativos y eficientes desde el punto de vista de las comunicaciones en entornos federados descentralizados (**RQ1**).

- O.3.** Diseñar y evaluar una solución técnica para DFL en escenarios dinámicos y adversariales, considerando información contextual y operacional para apoyar la fiabilidad a nivel descentralizado (**RQ2**).
- O.4.** Evaluar la robustez de la solución propuesta derivada de **O.3** frente a perturbaciones operativas, ataques de comunicación y ataques que afectan al proceso de aprendizaje colaborativo en un escenario adversarial acotado (**RQ2**).
- O.5.** Diseñar y evaluar una solución técnica basada en prototipos para DFL con datos multimodales heterogéneos, considerando distribuciones de clases non-IID y modalidades ausentes entre los participantes (**RQ3**).
- O.6.** Evaluar la eficiencia de las comunicaciones y el rendimiento del aprendizaje colaborativo en el intercambio de prototipos multimodales dentro de la solución propuesta en **O.5**, en comparación con enfoques basados en el intercambio de parámetros completos del modelo o actualizaciones completas del modelo (**RQ4**).

3 Metodología

Esta tesis doctoral se ha desarrollado mediante un enfoque científico que ha combinado una revisión exhaustiva del estado del arte, el diseño de soluciones técnicas y la evaluación de resultados a lo largo de etapas sucesivas de investigación. Dado que la tesis se presenta como un compendio de publicaciones, su metodología se articula a través de tres artículos de revista que examinan el mismo problema general desde ángulos complementarios. El primer artículo proporciona una caracterización fundacional de DFL y establece la base diagnóstica de la tesis mediante la identificación de las principales limitaciones, tendencias y retos abiertos del campo. A partir de este análisis, se han seleccionado dos retos especialmente relevantes para su posterior desarrollo técnico: la fiabilidad descentralizada bajo condiciones adversariales y dinámicas, y el aprendizaje colaborativo bajo heterogeneidad estadística y multimodal. En conjunto, estas publicaciones conforman una secuencia metodológica coherente que respalda el objetivo común de avanzar hacia un DFL robusto, adaptativo y eficiente en términos de comunicación en entornos heterogéneos.

La primera etapa abordó **RQ1** y los objetivos **O.1** y **O.2**, proporcionando una visión estructurada de DFL como paradigma de investigación emergente. Se llevó a cabo una revisión amplia de la literatura para identificar los aspectos que distinguen DFL de CFL, incluyendo arquitecturas de federación, topologías de red, mecanismos de comunicación, enfoques de seguridad y privacidad, indicadores clave de rendimiento, mecanismos de optimización, marcos de trabajo y escenarios de aplicación. La literatura revisada fue categorizada y analizada para establecer la estructura conceptual del campo, identificar mecanismos comúnmente utilizados, comparar los marcos de trabajo disponibles y determinar las tendencias, lecciones aprendidas y retos abiertos más relevantes. Este análisis proporcionó la terminología, la taxonomía y la base analítica necesarias para comprender por qué DFL necesita soluciones específicamente diseñadas para entornos descentralizados, dinámicos y heterogéneos. También estableció la etapa diagnóstica de la tesis, ya que las contribuciones técnicas posteriores se posicionan como respuestas a algunas de las limitaciones y oportunidades identificadas en este análisis fundacional. Estas consideraciones dieron lugar a la primera publicación de esta tesis doctoral, [Article 1–IEEE_COMST](#).

Tras completar el análisis del estado del arte e identificar las principales brechas en DFL, la metodología se centró entonces en la dimensión adversarial del aprendizaje descentralizado, abordando **RQ2** y los objetivos **O.3** y **O.4**. Esta transición estuvo motivada

por la necesidad de hacer DFL más fiable y robusto en entornos descentralizados realistas, donde la ausencia de un coordinador central desplaza la evaluación de la fiabilidad, las decisiones de agregación y las respuestas a perturbaciones hacia los nodos participantes. En esta etapa, la investigación estudió cómo podía reforzarse DFL en escenarios dinámicos en los que los participantes colaboran sin depender de una entidad coordinadora central y en los que los comportamientos maliciosos o poco fiables pueden afectar al proceso de aprendizaje. El reconocimiento aéreo militar se consideró un entorno de aplicación ideal para evaluar esta línea de investigación, ya que combina participantes móviles, conectividad cambiante, restricciones operativas predefinidas y posibles perturbaciones que afectan tanto al contexto de la misión como al proceso de aprendizaje colaborativo. En consecuencia, el diseño metodológico combinó el modelado de amenazas, el desarrollo de una solución técnica basada en la conciencia situacional y una evaluación empírica controlada, con el fin de investigar si la información contextual, operativa y relacionada con el modelo podía contribuir a una agregación descentralizada orientada a la fiabilidad en condiciones adversariales.

La evaluación de esta segunda etapa se llevó a cabo en un escenario acotado de reconocimiento aéreo militar utilizando datos de radar de apertura sintética (SAR), cuatro vehículos aéreos y diversas formaciones de vuelo bajo condiciones adversariales que afectaban tanto al contexto de la misión como al proceso de aprendizaje colaborativo. Este entorno permitió evaluar si los indicadores contextuales y los relacionados con el modelo podían apoyar la toma de decisiones de agregación descentralizada cuando los participantes estaban sujetos a restricciones de movilidad y comunicación, perturbaciones operativas y comportamientos potencialmente maliciosos [21]. Al combinar estas condiciones en un escenario controlado, esta etapa examinó cómo la conciencia situacional puede utilizarse para reforzar un DFL robusto y adaptativo bajo supuestos adversariales acotados. El trabajo resultante dio lugar a la segunda publicación de esta tesis doctoral, [Article 2–IEEE_COMMAG](#).

Una vez estudiadas las dimensiones adversariales y operacionales de DFL, la tercera etapa metodológica se centró en la heterogeneidad estadística y multimodal en entornos federados descentralizados, abordando **RQ3**, **RQ4** y los objetivos **O.5** y **O.6**. Esta etapa abordó escenarios en los que los clientes difieren en sus distribuciones locales de datos, las distribuciones de clases son non-IID y pueden faltar modalidades, mientras que el intercambio de parámetros del modelo o actualizaciones completas puede generar una sobrecarga de comunicación excesiva [30]. Para abordar este problema, se diseñó una solución técnica basada en prototipos para DFL multimodal. La metodología se basó en el intercambio de representaciones compactas de clase conscientes de la modalidad, en lugar de parámetros completos del modelo o de gradientes, lo que permitió que los participantes compartieran conocimiento con clientes vecinos de manera descentralizada, mientras se reducía la sobrecarga de comunicación [36]. En esta etapa, la robustez se caracterizó principalmente por la capacidad de preservar el rendimiento del aprendizaje colaborativo ante distribuciones heterogéneas de los datos de los clientes, la disponibilidad incompleta de modalidades y los recursos de comunicación limitados. Esta decisión metodológica vincula directamente la gestión de la heterogeneidad con la eficiencia comunicacional, dos dimensiones centrales para el objetivo general de la tesis.

El diseño metodológico de esta tercera contribución incorporó mecanismos para gestionar modalidades ausentes, alinear representaciones, ponderar las contribuciones de las modalidades y fusionar información multimodal en condiciones heterogéneas. También incluyó análisis teórico y validación experimental. El componente teórico examinó la convergencia y los costes de comunicación, mientras que el componente empírico evaluó el enfoque en *benchmarks* multimodales establecidos, configuraciones de datos non-IID, condiciones

de modalidades ausentes y comparaciones frente a baselines federados y descentralizados relevantes [29]. De este modo, la tercera etapa metodológica combinó el diseño de modelo, el intercambio descentralizado de conocimiento, el análisis formal y la evaluación empírica, sin asumir la presencia de una entidad central de agregación. Esta etapa dio lugar a la tercera publicación de esta tesis doctoral, [Article 3–ELSEVIER_IF](#).

En general, estas etapas metodológicas forman un hilo narrativo coherente que vincula la caracterización fundacional de DFL con dos dimensiones técnicas focalizadas: la fiabilidad adversarial mediante la conciencia situacional y el aprendizaje multimodal heterogéneo mediante el intercambio de prototipos eficiente en comunicaciones. Esta progresión proporciona la base metodológica para la siguiente sección, que sintetiza los resultados analíticos, empíricos y teóricos obtenidos de las tres publicaciones.

4 Resultados

Los resultados de esta tesis doctoral se sintetizan de acuerdo con los objetivos presentados en la Sección 2. Dado que la tesis se concibe como un compendio de publicaciones con una memoria integradora, los resultados no se derivan de una única aproximación experimental. En su lugar, emergen de una secuencia coherente de tres contribuciones científicas que abordan el DFL desde perspectivas complementarias. La primera contribución proporciona la base fundacional y analítica de la tesis al ofrecer una caracterización sistemática de DFL. Organiza el campo en torno a dimensiones arquitectónicas, de comunicación, de seguridad, de evaluación y de aplicación, e identifica tendencias, lecciones aprendidas y retos abiertos. La segunda contribución avanza desde el análisis conceptual hacia la evaluación técnica mediante el estudio de DFL en escenarios dinámicos y adversariales, utilizando una solución basada en la conciencia situacional. La tercera contribución aborda el DFL heterogéneo y multimodal mediante una solución basada en prototipos diseñada para operar bajo distribuciones de datos non-IID, modalidades ausentes y restricciones de comunicación. En conjunto, estos resultados respaldan el objetivo general de avanzar hacia un DFL robusto, adaptativo y eficiente a nivel de comunicaciones en entornos heterogéneos.

La publicación [Article 1–IEEE_COMST](#) presenta un estudio exhaustivo de DFL y la base conceptual necesaria para comprender el resto de la tesis. En la literatura revisada, la publicación también posicionó DFL como un tema que requiere un análisis dedicado, en lugar de abordarlo únicamente como un tema subsidiario dentro de estudios más amplios sobre CFL. El trabajo identificó y organizó los principales fundamentos que definen DFL como paradigma de investigación, incluidos las arquitecturas de federación, las topologías de red, los mecanismos de comunicación, los enfoques de seguridad y privacidad, los indicadores clave de rendimiento y los mecanismos de optimización. También revisó los marcos de trabajo disponibles, los escenarios de aplicación, las tendencias actuales, las lecciones aprendidas y los retos futuros. Esta contribución es relevante porque establece que DFL no puede entenderse únicamente como la eliminación del servidor central del FL clásico. En cambio, la descentralización redefine la distribución de responsabilidades entre los participantes e introduce nuevos retos de coordinación, comunicación, distribución de confianza, robustez, heterogeneidad y evaluación del rendimiento.

El survey identificó que DFL mitiga algunas limitaciones de CFL, particularmente la dependencia de una entidad central de agregación, la presencia de un punto único de fallo y los cuellos de botella asociados a la coordinación basada en servidor. Sin embargo, también plantea retos técnicos específicos. Estos incluyen la necesidad de gestionar la agregación descentralizada, mantener la comunicación entre los participantes, manejar datos y dispositivos heterogéneos, definir topologías adecuadas, evaluar el rendimiento mediante

KPIs apropiados y abordar cuestiones de seguridad y privacidad sin depender de una autoridad central de confianza. Por tanto, un resultado central de [Article 1–IEEE_COMST](#) es la identificación de DFL como un paradigma cuya adopción práctica requiere mecanismos específicamente diseñados para entornos descentralizados y heterogéneos, en lugar de adaptaciones directas de los supuestos de CFL.

Esta primera contribución respalda **O.1** y **O.2**, que, conjuntamente, proporcionan la base para responder **RQ1**. En relación con **O.1**, ofrece un análisis estructurado del estado del arte en DFL y define los elementos conceptuales y técnicos empleados a lo largo de la tesis. Con respecto a **O.2**, identifica las limitaciones, tendencias y retos abiertos que motivan las contribuciones técnicas posteriores. En particular, el survey explica por qué la robustez, la adaptación, la eficiencia a nivel de comunicaciones y la gestión de la heterogeneidad son cuestiones centrales en DFL.

Los resultados de [Article 1–IEEE_COMST](#) contribuyen, además, al hilo narrativo global de la tesis al identificar los principales escenarios de aplicación en los que DFL puede resultar relevante. Estos escenarios incluyen el sector sanitario, la Industria 4.0, los servicios móviles, los entornos militares y los sistemas vehiculares, entre otros. La relevancia de estos escenarios no radica solo en su diversidad, sino también en las restricciones recurrentes que exponen: los datos están distribuidos, la centralización puede ser inviable o impracticable, los participantes pueden ser heterogéneos y las condiciones de comunicación pueden variar con el tiempo. En el contexto de esta tesis, este análisis a nivel de escenarios sirve de puente entre el survey fundacional y las contribuciones técnicas posteriores. Muestra que los retos identificados en [Article 1–IEEE_COMST](#) pueden instanciarse en dos direcciones de investigación complementarias: la fiabilidad descentralizada bajo condiciones dinámicas y adversariales, abordada en [Article 2–IEEE_COMMAG](#), y el aprendizaje eficiente a nivel de comunicaciones ante la disponibilidad heterogénea de datos multimodales, abordado en [Article 3–ELSEVIER_IF](#). Por esta razón, el escenario adversarial de reconocimiento aéreo considerado en [Article 2–IEEE_COMMAG](#) y el escenario de aprendizaje multimodal heterogéneo considerado en [Article 3–ELSEVIER_IF](#) no se presentan como casos de estudio aislados, sino como respuestas dirigidas a dos retos principales identificados en el análisis fundacional.

La publicación [Article 2–IEEE_COMMAG](#) se centra en la robustez y la adaptación de DFL en escenarios dinámicos y adversariales. Propone *Flighter*, una solución de conciencia situacional para DFL, y la evalúa en un escenario adversarial acotado de reconocimiento aéreo. La publicación considera varias aeronaves que colaboran para entrenar modelos de detección de vehículos enemigos sin depender de un servidor central de coordinación. Este entorno es exigente porque los participantes son móviles, los enlaces de comunicación pueden ser inestables y los adversarios pueden afectar tanto el contexto operativo como el proceso de aprendizaje colaborativo [20]. El resultado principal de esta publicación es que la conciencia situacional puede integrarse en DFL para apoyar la toma de decisiones descentralizadas sobre la fiabilidad durante el entrenamiento colaborativo.

Flighter incorpora información contextual, operativa y relacionada con el modelo en el proceso de defensa. Su módulo de defensa combina la similitud de los modelos con indicadores de conciencia situacional: la formación de vuelo, el geoposicionamiento, el comportamiento de interconexión y el uso de recursos. Estos indicadores reflejan limitaciones previamente identificadas para DFL en entornos dinámicos y adversariales, en los que los participantes pueden ser móviles, los enlaces de comunicación pueden ser inestables, el comportamiento operativo puede desviarse de los patrones esperados y las contribuciones maliciosas pueden afectar a la agregación descentralizada. Los indicadores se normalizan y se combinan mediante una puntuación de defensa ponderada que representa la fiabilidad estimada de cada participante. Esta puntuación se evalúa en función de un umbral dinámico

que se ajusta durante el proceso de entrenamiento. Los participantes cuyas puntuaciones superan el umbral reciben una mayor influencia en la agregación descentralizada, mientras que se reducen los pesos de agregación de quienes se sitúan por debajo del umbral. Si la baja fiabilidad persiste durante varias rondas, las conexiones correspondientes pueden interrumpirse para proteger el proceso de aprendizaje colaborativo. Este diseño aborda **O.3** y contribuye a responder **RQ2**, al mostrar cómo una solución basada en la conciencia situacional puede favorecer una agregación descentralizada orientada a la fiabilidad en un entorno dinámico y adversarial. En consecuencia, *Flighter* se considera en esta tesis una contribución acotada a la robustez y la adaptación en DFL adversarial, y no un mecanismo de defensa de propósito general aplicable a cualquier entorno federado descentralizado.

Los resultados empíricos de [Article 2–IEEE_COMMAG](#) respaldan **O.4** y, además, contribuyen a **RQ2**. La publicación evaluó *Flighter* en un escenario simulado de reconocimiento aéreo militar que involucraba cuatro aeronaves equipadas con SAR, varias formaciones de vuelo y condiciones adversas que afectaban el geoposicionamiento, el comportamiento de la trayectoria y el entrenamiento colaborativo. Los resultados cuantitativos reportados en el artículo corresponden a la evaluación basada en VGG16 sobre tres conjuntos de datos SAR: *MSTAR* para objetivos terrestres militares [37], *SAMPLE* para pares de objetivos SAR medidos y sintéticos [38], y *OpenSARShip* para la clasificación de barcos marítimos mediante SAR [39]. La comparación se realizó frente a una línea base DFL del estado del arte que replicaba el proceso de entrenamiento, la arquitectura del modelo, el procesamiento de datos y las rondas de federación, pero excluía el módulo de defensa basado en la conciencia situacional. En el escenario base de reconocimiento, *Flighter* alcanzó valores de F1-score de 95.8% en *MSTAR*, 97.5% en *SAMPLE* y 79.1% en *OpenSARShip*. El uso de recursos se mantuvo moderado en esta configuración, con un uso de CPU alrededor del 57% y un uso de RAM de alrededor de 1.7 GB, mientras que la pérdida de paquetes y los tiempos de ronda se mantuvieron estables, alrededor de 0.7% de pérdida de paquetes y 5.3 s por ronda en la configuración base. A pesar de las desviaciones en el geoposicionamiento, *Flighter* mantuvo valores de F1-score de 91.8% en *MSTAR*, 90.3% en *SAMPLE* y 72.0% en *OpenSARShip*, con una pérdida de paquetes inferior al 1% y solo ligeros aumentos en el tiempo de ronda. La evaluación también reportó valores de F1-score de 92.5%, 89.8% y 70.1% al manipular la trayectoria de colisión en *MSTAR*, *SAMPLE* y *OpenSARShip*, respectivamente. Por el contrario, los escenarios adversariales de envenenamiento de datos produjeron valores de F1-score que oscilaban entre 52.1% y 90.1%, dependiendo del dataset y de la configuración del ataque. Estos resultados deben interpretarse en el contexto de los datasets, ataques, arquitectura de modelo, línea base y condiciones de simulación específicos reportados en la publicación.

La publicación [Article 3–ELSEVIER_IF](#) aborda el DFL heterogéneo y multimodal. Examina escenarios de aprendizaje colaborativo en los que los participantes presentan distribuciones de datos non-IID, disponibilidad desigual de modalidades, modalidades ausentes y restricciones de comunicación. Para abordar estos retos, la publicación propone *Modalis*, una solución basada en prototipos para DFL multimodal, construida en torno a un protocolo de intercambio de prototipos eficientes a nivel de comunicaciones. En lugar de transmitir parámetros completos del modelo o actualizaciones de alta dimensionalidad, *Modalis* permite a los participantes intercambiar prototipos de clase compactos y conscientes de la modalidad que resumen información discriminativa de clases aprendida a partir de datos locales heterogéneos. Este diseño es central para la tesis porque conecta dos retos principales identificados en el survey: la gestión de la heterogeneidad y la eficiencia de las comunicaciones. Además de la validación empírica, la publicación aporta soporte teórico mediante un resultado de convergencia y un análisis del coste de comunicación,

enmarcando el intercambio de prototipos como un mecanismo que contribuye tanto al aprendizaje colaborativo multimodal como a la colaboración descentralizada y eficiente a nivel de comunicaciones.

Más allá del protocolo de comunicación, Modalis incorpora mecanismos para manejar modalidades ausentes, alinear representaciones multimodales, ponderar las contribuciones de las modalidades y fusionar la información ante la disponibilidad heterogénea de datos. En particular, los *embeddings* contextuales nulos proporcionan representaciones sustitutivas cuando una modalidad está ausente, el adaptive gating ajusta la relevancia de las modalidades disponibles y el componente de fusión multimodal integra información parcial en una representación compartida. Estos mecanismos se complementan con objetivos de alineación de representaciones que promueven la consistencia entre los *embeddings* multimodales locales y los prototipos intercambiados. Como resultado, los participantes pueden beneficiarse de la colaboración descentralizada incluso cuando sus datos locales son incompletos, están sesgados estadísticamente o se limitan a un subconjunto de las modalidades disponibles en la federación. Este resultado respalda **O.5** y contribuye a responder **RQ3** al mostrar que un enfoque basado en prototipos puede diseñarse y evaluarse para DFL con distribuciones de clases non-IID y modalidades ausentes entre participantes, sin depender de una entidad central para realizar la fusión multimodal ni para manejar las modalidades ausentes.

Los resultados experimentales reportados en [Article 3-ELSEVIER_IF](#) muestran una mejora del rendimiento en condiciones de DFL multimodal heterogéneo. La evaluación consideró cuatro datasets multimodales: AVMNIST para reconocimiento audiovisual de dígitos hablados [40], CREMA-D para análisis audiovisual de habla emocional [41], UCI-HAR para reconocimiento de actividad humana mediante datos de acelerómetro y giroscopio [42], y CRISIS-MMD para clasificación de eventos de crisis en redes sociales mediante imagen-texto [43]. El sesgo de clases non-IID se controló mediante $\beta \in 1.0, 0.5, 0.1$, y las configuraciones de modalidades ausentes se controlaron mediante $p_m \in 0.0, 0.25, 0.5$. Los experimentos involucraron a 20 clientes conectados mediante un grafo estático de Erdős-Rényi con $p_{ER} = 0.9$ para los resultados principales reportados, y Modalis se comparó frente a una cota superior centralizada y varias líneas base FL/DFL, incluidas FedAvg, FedProto, FedInMM, CreamFL, PmcmFL, FedMAC y MFCPL [31, 28, 29, 34]. Bajo alta heterogeneidad, el artículo reporta una ganancia relativa de F1-score de hasta aproximadamente 4.0% sobre el siguiente método competidor con mejor rendimiento. Los resultados también muestran que la degradación del rendimiento aumenta a medida que el sesgo de etiquetas y la información ausente se vuelven más severos, aunque Modalis generalmente presenta pérdidas relativas de rendimiento menores que las de los métodos comparados.

Un resultado particularmente relevante se refiere a los clientes con una sola modalidad disponible. En la configuración más desafiante analizada en el artículo, Modalis apoyó la transferencia de conocimiento cross-modal para clientes unimodales, alcanzando un F1-score de 82.1% en AVMNIST para clientes con solo imagen, lo que corresponde a una ganancia relativa del 19.9% sobre FedAvg. Este resultado sugiere que el intercambio de prototipos puede ayudar a los participantes descentralizados a beneficiarse de conocimiento asociado a modalidades no disponibles localmente, aunque esta interpretación sigue estando condicionada por los datasets, las configuraciones de modalidades, las líneas base y los supuestos experimentales documentados en la publicación. El artículo también reporta un análisis de transferibilidad entre datasets en el que una configuración de Modalis ajustada sobre AVMNIST incurrió solo en descensos modestos de F1-score en CREMA-D, UCI-HAR y CRISIS-MMD, manteniendo al mismo tiempo las mejores puntuaciones absolutas entre los métodos comparados. Estos resultados respaldan la robustez frente a la transferencia de

hiperparámetros bajo las condiciones evaluadas, pero no implican generalización zero-shot entre datasets incompatibles.

Modalis también respalda **O.6** y contribuye a responder **RQ4** mediante sus resultados sobre la eficiencia comunicacional. En la configuración AVMNIST computacionalmente intensiva con alto sesgo de etiquetas y modalidades ausentes, Modalis reportó un coste medio de comunicación de aproximadamente 0.11 MB por mensaje de cliente y un F1-score de 83.4%. Esta carga de comunicación es más de 40× menor que la de métodos con elevada carga de comunicaciones como FedAvg y PmcmFL, que transmiten alrededor de 4.75–4.80 MB por ronda. Este resultado es central para la tesis porque la eficiencia en las comunicaciones es un requisito estructural para DFL: en entornos descentralizados, el intercambio excesivo de parámetros del modelo puede limitar la escalabilidad, aumentar la latencia o reducir la aplicabilidad en escenarios con recursos limitados. El análisis de recursos también reportó aproximadamente un 44% de utilización de CPU, un 70% de utilización de GPU y un 31% de utilización de RAM en el nodo de evaluación, situando a Modalis cerca de líneas base avanzadas de FL multimodal en términos de huella computacional.

En conjunto, las tres publicaciones proporcionan la base analítica y empírica para abordar el problema planteado desde perspectivas complementarias. [Article 1–IEEE_COMST](#) establece la estructura conceptual de DFL e identifica los retos que motivan el trabajo técnico posterior. [Article 2–IEEE_COMMAG](#) demuestra, en un escenario adversarial acotado de reconocimiento aéreo, cómo la conciencia situacional puede apoyar la agregación descentralizada orientada a la fiabilidad en medio de la movilidad, la inestabilidad de las comunicaciones y el comportamiento adversarial. [Article 3–ELSEVIER_IF](#) muestra cómo el intercambio de prototipos multimodales puede apoyar el aprendizaje colaborativo con datos non-IID, modalidades ausentes y restricciones de comunicación. Estos resultados conectan la caracterización fundacional de DFL con dos respuestas técnicas dirigidas a los retos de robustez, heterogeneidad y eficiencia comunicacional en entornos federados descentralizados.

5 Conclusiones y trabajo futuro

Esta tesis doctoral ha estudiado cómo avanzar DFL hacia un aprendizaje colaborativo más robusto en entornos heterogéneos, adversariales y restringidos por la comunicación, mediante tres publicaciones principales en revistas que abordan cuatro preguntas de investigación complementarias. **RQ1** examinó los fundamentos, marcos de trabajo, escenarios de aplicación, métricas, limitaciones y retos que definen el DFL. **RQ2** se centró en cómo la fiabilidad y la robustez pueden reforzarse frente al comportamiento adversarial, a las perturbaciones operativas y a los nodos potencialmente comprometidos. **RQ3** abordó el aprendizaje multimodal descentralizado bajo condiciones de datos non-IID y modalidades ausentes, mientras que **RQ4** consideró el efecto de los mecanismos eficientes de intercambio sobre el coste de comunicación y el rendimiento del aprendizaje colaborativo. Estas preguntas se abordaron mediante tres contribuciones complementarias: una caracterización fundacional del DFL [Article 1–IEEE_COMST](#); una solución basada en la conciencia situacional, evaluada en un escenario adversarial acotado [Article 2–IEEE_COMMAG](#); y una solución basada en prototipos para el DFL multimodal heterogéneo [Article 3–ELSEVIER_IF](#). En conjunto, estas contribuciones respaldan el argumento central de la tesis: el DFL puede mitigar algunas limitaciones estructurales del CFL, pero su adopción práctica requiere mecanismos diseñados específicamente para entornos descentralizados, heterogéneos y dinámicos, en los que participan actores potencialmente adversariales.

Con respecto a **RQ1**, la tesis concluye que el DFL constituye una forma específica

de aprendizaje colaborativo que reduce la necesidad de compartir datos sin procesar, en la que la ausencia de una entidad central de agregación cambia la manera en que deben diseñarse la coordinación, la comunicación, la agregación, la confianza y la evaluación. El análisis realizado en [Article 1–IEEE_COMST](#) muestra que la descentralización redistribuye responsabilidades a lo largo de la federación, incluyendo la comunicación, la agregación, la gestión de la topología, la robustez, la privacidad, la distribución de la confianza y la evaluación del rendimiento. Esta contribución aborda **O.1** y **O.2** al organizar los principales fundamentos del DFL e identificar las limitaciones, tendencias, lecciones aprendidas y retos abiertos que motivan el desarrollo de mecanismos robustos, adaptativos y eficientes en comunicación para entornos heterogéneos y dinámicos.

Para **RQ2**, la tesis concluye que, dentro de las condiciones adversas acotadas evaluadas, la combinación de información relacionada con el modelo con el conocimiento del contexto y del funcionamiento puede favorecer una agregación descentralizada orientada a la fiabilidad y mejorar la robustez frente a las perturbaciones consideradas. La contribución desarrollada en [Article 2–IEEE_COMMAG](#) propuso *Flighter* como solución técnica para escenarios DFL dinámicos y adversariales. Al incorporar indicadores como la similitud del modelo, la formación de vuelo, el geoposicionamiento, el uso de recursos y la tasa de interconexión, *Flighter* apoya la toma de decisiones de agregación orientadas a la fiabilidad sin depender de un supervisor central. Su evaluación en un escenario acotado de reconocimiento aéreo basado en SAR indica que la conciencia situacional puede apoyar la robustez descentralizada bajo las condiciones adversariales consideradas. Esta contribución aborda **O.3** y **O.4**, manteniendo el entorno militar como escenario de evaluación en lugar del alcance temático general de la tesis.

En relación con **RQ3** y **RQ4**, la tesis concluye que el intercambio de conocimiento basado en prototipos es un mecanismo adecuado para apoyar el aprendizaje multimodal descentralizado bajo condiciones de datos non-IID y con modalidades ausentes, reduciendo al mismo tiempo los costes de comunicación en los escenarios evaluados. La contribución presentada en [Article 3–ELSEVIER_IF](#) propuso *Modalis*, una solución técnica basada en prototipos que sustituye el intercambio completo de parámetros del modelo por prototipos de clase compactos y conscientes de la modalidad. Estas representaciones permiten a los participantes compartir conocimiento discriminativo con clientes vecinos, al tiempo que facilitan la gestión de modalidades ausentes, el alineamiento de representaciones, la ponderación de modalidades y la fusión de información multimodal. Los resultados reportados indican que *Modalis* mejora el aprendizaje colaborativo en las configuraciones heterogéneas evaluadas y reduce los costes de comunicación en comparación con enfoques basados en el intercambio de modelos completos. Esta contribución aborda **O.5** y **O.6** al vincular la gestión de la heterogeneidad multimodal con la colaboración descentralizada y eficiente en la comunicación.

El alcance de la tesis también queda definido por varias limitaciones. En primer lugar, el análisis fundacional realizado en [Article 1–IEEE_COMST](#) está necesariamente limitado temporalmente, dado que DFL es un área de investigación activa en la que continúan surgiendo nuevos marcos de trabajo, estrategias de optimización, mecanismos de privacidad y escenarios de aplicación. Además, aunque el trabajo examinó la seguridad y la privacidad como dimensiones fundamentales del DFL, no realizó un análisis específico centrado en los adversarios, las superficies de ataque y las relaciones entre ataques y defensas, aspectos que posteriormente han recibido una atención más especializada [8]. Por tanto, su taxonomía, tendencias, lecciones aprendidas y retos abiertos deben entenderse como una caracterización estructurada del campo en el momento de la publicación, y no como una descripción definitiva o exhaustiva del DFL. En segundo lugar, las dos contribuciones técnicas, [Article 2–](#)

[IEEE_COMMAG](#) y [Article 3–ELSEVIER_IF](#), fueron evaluadas en condiciones controladas, lo que favorece la claridad analítica y la reproducibilidad, pero limita su generalización directa a despliegues operativos. En el caso de *Flighter*, las principales limitaciones se centran en la validez externa más allá del entorno simulado, la escalabilidad a federaciones más grandes y más heterogéneas, y la cobertura de estrategias adversariales, regímenes de comunicación y perturbaciones operativas adicionales. En el caso de *Modalis*, las principales limitaciones se refieren a la evaluación basada en *benchmarks*, la generación controlada de heterogeneidad y las suposiciones predefinidas sobre la comunicación descentralizada. Sería necesaria una validación adicional para evaluar su comportamiento ante flujos de datos naturalmente heterogéneos, topologías dinámicas, capacidades heterogéneas de los dispositivos, restricciones energéticas y condiciones reales de red *edge*. Finalmente, aunque el intercambio de prototipos reduce los costes de comunicación al evitar la transmisión de los parámetros completos del modelo, las representaciones compactas no deben interpretarse como una protección formal de la privacidad por sí mismas. Un análisis dedicado a la privacidad, que podría incluir privacidad diferencial, agregación segura o mecanismos de evaluación de la fuga de información, queda fuera del alcance empírico de la presente tesis. Estas limitaciones condicionan, en cierto modo, las aproximaciones metodológicas y empíricas a partir de las cuales deben interpretarse las conclusiones.

Los resultados de esta tesis abren varias líneas de investigación futuras. Una primera dirección es evaluar soluciones DFL bajo condiciones de despliegue más cercanas a las esperadas en sistemas descentralizados reales. Las evaluaciones realizadas en esta tesis proporcionan evidencia controlada sobre escenarios adversariales y heterogéneos específicos, pero los despliegues reales pueden implicar un mayor número de participantes, relaciones dinámicas de vecindad, disponibilidad intermitente, capacidades computacionales heterogéneas y enlaces de comunicación inestables. Estos factores no deben tratarse como fuentes aisladas de degradación, dado que los cambios en topología, disponibilidad de participantes, calidad de comunicación y recursos de los dispositivos pueden afectar conjuntamente a la convergencia, la robustez y la eficiencia a nivel de comunicaciones. El trabajo futuro debería, por tanto, analizar cómo se comportan los mecanismos DFL cuando estas restricciones evolucionan simultáneamente durante el entrenamiento, especialmente en entornos colaborativos móviles y orientados al *edge*, donde no se puede asumir que la federación permanezca estable a lo largo del tiempo.

Una segunda dirección consiste en ampliar la conciencia situacional como mecanismo de fiabilidad para la colaboración descentralizada. En *Flighter*, los indicadores contextuales y operacionales se utilizan para apoyar decisiones de agregación orientadas a la fiabilidad en condiciones adversariales acotadas. Un siguiente paso relevante es estudiar cómo deberían seleccionarse, ponderarse, actualizarse y validarse dichos indicadores cuando el contexto operativo cambia o la información contextual disponible es incompleta, presenta ruido o está potencialmente manipulada. Abordar esta línea de investigación requiere ir más allá de los indicadores y supuestos específicos evaluados en la tesis para examinar si las puntuaciones de fiabilidad siguen siendo significativas cuando los participantes difieren en patrones de movilidad, calidad de la comunicación, recursos computacionales y consistencia del comportamiento. La investigación futura también debería analizar cuánta información contextual es necesaria para mejorar la robustez sin introducir una sobrecarga de comunicación excesiva ni nuevas superficies de ataque.

Una tercera línea de trabajo futura es el análisis formal de la privacidad del DFL basado en prototipos. *Modalis* reduce los costes de comunicación mediante el intercambio de prototipos de clase compactos y conscientes de la modalidad en lugar de parámetros completos del modelo, pero no debe asumirse que estas representaciones compactas propor-

cionen garantías de privacidad por sí mismas. Los prototipos pueden seguir codificando información sobre distribuciones locales de clases, representaciones de características o patrones de disponibilidad de modalidades, especialmente cuando los clientes disponen de conjuntos de datos pequeños o muy sesgados. La investigación futura debería cuantificar la posible fuga de información asociada al intercambio de prototipos y estudiar mecanismos de preservación de la privacidad adaptados específicamente al aprendizaje descentralizado mediante prototipos. Estas investigaciones podrían incluir mecanismos de privacidad diferencial, métodos de construcción de prototipos orientados a preservar la privacidad, agregación segura entre nodos vecinos o métodos empíricos de evaluación de filtraciones de información que determinen qué podría inferir un adversario a partir de las representaciones intercambiadas.

En resumen, esta tesis doctoral avanza en el estudio del DFL robusto en entornos heterogéneos mediante una secuencia coherente de contribuciones analíticas y técnicas. La tesis muestra que el DFL puede reducir la dependencia de una entidad central de agregación, al tiempo que evidencia que su adopción práctica requiere mecanismos diseñados para abordar los retos derivados de la descentralización. Al caracterizar el campo, diseñar y evaluar una solución técnica basada en la conciencia situacional para escenarios adversariales acotados, y desarrollar una solución técnica basada en prototipos para el aprendizaje multimodal heterogéneo, la tesis contribuye a una comprensión precisa y fundamentada empíricamente de cómo el DFL puede evolucionar hacia un aprendizaje colaborativo robusto, adaptativo y eficiente en la comunicación.

Bibliography

- [1] E. Dritsas and M. Trigka, “Federated learning for IoT: A survey of techniques, challenges, and applications,” *Journal of Sensor and Actuator Networks*, vol. 14, no. 1, p. 9, 2025. DOI: 10.3390/jsan14010009
- [2] S. Alahmari and I. Alghamdi, “A comprehensive survey on energy-efficient and privacy-preserving federated learning for edge intelligence and IoT,” *Results in Engineering*, vol. 28, p. 107849, 2025. DOI: 10.1016/j.rineng.2025.107849
- [3] L. I. Barona López and T. Borja Saltos, “Heterogeneity challenges of federated learning for future wireless communication networks,” *Journal of Sensor and Actuator Networks*, vol. 14, no. 2, p. 37, 2025. DOI: 10.3390/jsan14020037
- [4] P. Kairouz *et al.*, “Advances and open problems in federated learning,” *Foundations and Trends in Machine Learning*, vol. 14, no. 1–2, pp. 1–210, 2021. DOI: 10.1561/22000000083
- [5] M. Arbaoui, M. E. A. Brahmia, A. Rahmoun, and M. Zghal, “Federated learning survey: A multi-level taxonomy of aggregation techniques, experimental insights, and future frontiers,” *ACM Transactions on Intelligent Systems and Technology*, vol. 15, no. 6, pp. 1–69, 2024. DOI: 10.1145/3678182
- [6] H. B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. Agüera y Arcas, “Communication-efficient learning of deep networks from decentralized data,” in *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, ser. Proceedings of Machine Learning Research, vol. 54. PMLR, 2017, pp. 1273–1282.
- [7] L. Yuan, Z. Wang, L. Sun, P. S. Yu, and C. G. Brinton, “Decentralized federated learning: A survey and perspective,” *IEEE Internet of Things Journal*, vol. 11, no. 21, pp. 34 617–34 638, 2024. DOI: 10.1109/JIOT.2024.3407584
- [8] E. Hallaji, R. Razavi-Far, M. Saif, B. Wang, and Q. Yang, “Decentralized federated learning: A survey on security and privacy,” *IEEE Transactions on Big Data*, vol. 10, no. 2, pp. 194–213, 2024. DOI: 10.1109/TBDATA.2024.3362191
- [9] Y. Zhou, M. Shi, Y. Tian, Q. Ye, and J. Lv, “DeFTA: A plug-and-play peer-to-peer decentralized federated learning framework,” *Information Sciences*, vol. 670, p. 120582, 2024. DOI: 10.1016/j.ins.2024.120582

- [10] E. Fragkou and D. Katsaros, “A joint survey in decentralized federated learning and TinyML: A brief introduction to swarm learning,” *Future Internet*, vol. 16, no. 11, p. 413, 2024. DOI: 10.3390/fi16110413
- [11] W.-C. Chung, C.-A. Lo, Y.-H. Lin, Z.-H. Chen, and C.-L. Hung, “Decentralized federated learning with non-IID data: Challenges, trends, and future opportunities,” *ACM Computing Surveys*, vol. 58, no. 8, pp. 1–41, 2026. DOI: 10.1145/3785657
- [12] K. N. Kumar, C. K. Mohan, and L. R. Cenkeramaddi, “The impact of adversarial attacks on federated learning: A survey,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 46, no. 5, pp. 2672–2691, 2024. DOI: 10.1109/TPAMI.2023.3322785
- [13] J. C. Zhao *et al.*, “The federation strikes back: A survey of federated learning privacy attacks, defenses, applications, and policy landscape,” *ACM Computing Surveys*, vol. 57, no. 9, pp. 1–37, 2025. DOI: 10.1145/3724113
- [14] L. Bai, H. Hu, Q. Ye, H. Li, L. Wang, and J. Xu, “Membership inference attacks and defenses in federated learning: A survey,” *ACM Computing Surveys*, vol. 57, no. 4, pp. 1–35, 2025. DOI: 10.1145/3704633
- [15] Y. Feng *et al.*, “A survey of security threats in federated learning,” *Complex & Intelligent Systems*, vol. 11, p. 165, 2025. DOI: 10.1007/s40747-024-01664-0
- [16] H. U. Manzoor, T. H. H. Aldhyani, M. Y. Alzahrani, and N. Alsubait, “A survey of security strategies in federated learning,” *Future Internet*, vol. 16, no. 10, p. 374, 2024. DOI: 10.3390/fi16100374
- [17] S. MahmoudZadeh, A. Yazdani, Y. Kalantari, B. Ciftler, F. Aidarus, and M. O. Al Kadri, “Holistic review of UAV-centric situational awareness: Applications, limitations, and algorithmic challenges,” *Robotics*, vol. 13, no. 8, p. 117, 2024. DOI: 10.3390/robotics13080117
- [18] M. R. Endsley, “Toward a theory of situation awareness in dynamic systems,” *Human Factors*, vol. 37, no. 1, pp. 32–64, 1995. DOI: 10.1518/001872095779049543
- [19] H. Li and J. Huang, “Hierarchical federated deep reinforcement learning based joint communication and computation for UAV situation awareness,” *Vehicular Communications*, vol. 50, p. 100853, 2024. DOI: 10.1016/j.vehcom.2024.100853
- [20] Y. Qu *et al.*, “Decentralized federated learning for UAV networks: Architecture, challenges, and opportunities,” *IEEE Network*, vol. 35, no. 6, pp. 156–162, 2021. DOI: 10.1109/MNET.001.2100253
- [21] W. Yang *et al.*, “DFUN-KDF: Efficient and robust decentralized federated framework for UAV networks via knowledge distillation and filtering,” *Communications in Transportation Research*, vol. 5, p. 100173, 2025. DOI: 10.1016/j.commtr.2025.100173
- [22] M. Ye, X. Fang, B. Du, P. C. Yuen, and D. Tao, “Heterogeneous federated learning: State-of-the-art and research challenges,” *ACM Computing Surveys*, vol. 56, no. 3, pp. 1–44, 2023. DOI: 10.1145/3625558
- [23] Z. Lu, H. Pan, Y. Dai, X. Si, and Y. Zhang, “Federated learning with non-IID data: A survey,” *IEEE Internet of Things Journal*, vol. 11, no. 11, pp. 19 188–19 209, 2024. DOI: 10.1109/JIOT.2024.3376548

- [24] H. Zhu, J. Xu, S. Liu, and Y. Jin, “Federated learning on non-IID data: A survey,” *Neurocomputing*, vol. 465, pp. 371–390, 2021. DOI: 10.1016/j.neucom.2021.07.098
- [25] L. Che, J. Wang, Y. Zhou, and F. Ma, “Multimodal federated learning: A survey,” *Sensors*, vol. 23, no. 15, p. 6986, 2023. DOI: 10.3390/s23156986
- [26] Y.-M. Lin, Y. Gao, M.-G. Gong, Y. Wang, and Z.-Y. Li, “Federated learning on multimodal data: A comprehensive survey,” *Machine Intelligence Research*, vol. 20, no. 4, pp. 539–553, 2023. DOI: 10.1007/s11633-022-1398-0
- [27] H. Pan, X. Zhao, L. He, Y. Shi, and X. Lin, “A survey of multimodal federated learning: Background, applications, and perspectives,” *Multimedia Systems*, vol. 30, no. 4, p. 222, 2024. DOI: 10.1007/s00530-024-01422-9
- [28] S. Yu, J. Wang, W. Hussein, and P. C. K. Hung, “Robust multimodal federated learning for incomplete modalities,” *Computer Communications*, vol. 214, pp. 234–243, 2024. DOI: 10.1016/j.comcom.2023.12.003
- [29] K. Yin, Z. Ding, X. Ji, and Z. Wang, “Self-attention fusion and adaptive continual updating for multimodal federated learning with heterogeneous data,” *Neural Networks*, vol. 187, p. 107345, 2025. DOI: 10.1016/j.neunet.2025.107345
- [30] W. Huang, D. Wang, X. Ouyang, J. Wan, J. Liu, and T. Li, “Multimodal federated learning: Concept, methods, applications and future directions,” *Information Fusion*, vol. 112, p. 102576, 2024. DOI: 10.1016/j.inffus.2024.102576
- [31] Y. Tan *et al.*, “FedProto: Federated prototype learning across heterogeneous clients,” in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 36, no. 8, 2022, pp. 8432–8440. DOI: 10.1609/aaai.v36i8.20819
- [32] X. Mu *et al.*, “FedProc: Prototypical contrastive federated learning on non-IID data,” *Future Generation Computer Systems*, vol. 143, pp. 93–104, 2023. DOI: 10.1016/j.future.2023.01.019
- [33] L. Qi, H. Chen, H. Zou, S. Chen, X. Zhang, and H. Chen, “Decentralized federated learning with prototype exchange,” *Mathematics*, vol. 13, no. 2, p. 237, 2025. DOI: 10.3390/math13020237
- [34] H. Q. Le *et al.*, “Cross-modal prototype based multimodal federated learning under severely missing modality,” *Information Fusion*, vol. 122, p. 103219, 2025. DOI: 10.1016/j.inffus.2025.103219
- [35] M. D. Nguyen, T. T. Nguyen, H. H. Pham, T. N. Hoang, P. L. Nguyen, and T. T. Huynh, “FedMAC: Tackling partial-modality missing in federated learning with cross-modal aggregation and contrastive regularization,” in *Proceedings of the 2024 IEEE 23rd International Symposium on Network Computing and Applications*. IEEE, 2024. DOI: 10.1109/NCA63107.2024.00047
- [36] B. Li, W. Gao, J. Xie, M. Gong, L. Wang, and H. Li, “Prototype-based decentralized federated learning for the heterogeneous time-varying IoT systems,” *IEEE Internet of Things Journal*, vol. 11, no. 4, pp. 6916–6927, 2024. DOI: 10.1109/JIOT.2023.3313118
- [37] T. D. Ross, S. W. Worrell, V. J. Velten, J. C. Mossing, and M. L. Bryant, “Standard sar atr evaluation experiments using the mstar public release data set,” in *Algorithms*

- for *Synthetic Aperture Radar Imagery V*, vol. 3370. SPIE, 1998, pp. 566–573. DOI: 10.1117/12.321859
- [38] B. Lewis, T. Scarnati, E. Sudkamp, J. W. Nehrbass, S. Rosencrantz, and E. G. Zelnio, “A sar dataset for atr development: the synthetic and measured paired labeled experiment (SAMPLE),” in *Algorithms for Synthetic Aperture Radar Imagery XXVI*, vol. 10987. SPIE, 2019, pp. 39–54. DOI: 10.1117/12.2523460
 - [39] L. Huang *et al.*, “OpenSARShip: A dataset dedicated to sentinel-1 ship interpretation,” *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 11, no. 1, pp. 195–208, 2018. DOI: 10.1109/JSTARS.2017.2755672
 - [40] S. Becker, J. Vielhaben, M. Ackermann, K.-R. Müller, S. Lapuschkin, and W. Samek, “Audiomnist: Exploring explainable artificial intelligence for audio analysis on a simple benchmark,” *Journal of the Franklin Institute*, vol. 361, no. 1, pp. 418–428, 2024. DOI: 10.1016/j.jfranklin.2023.11.038
 - [41] H. Cao, D. G. Cooper, M. K. Keutmann, R. C. Gur, A. Nenkova, and R. Verma, “Cremad: Crowd-sourced emotional multimodal actors dataset,” *IEEE Transactions on Affective Computing*, vol. 5, no. 4, pp. 377–390, 2014. DOI: 10.1109/TAFFC.2014.2336244
 - [42] D. Anguita, A. Ghio, L. Oneto, X. Parra, J. L. Reyes-Ortiz *et al.*, “A public domain dataset for human activity recognition using smartphones,” in *Proceedings of the 21th International European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning*, vol. 3, 2013, pp. 3–4.
 - [43] F. Alam, F. Ofli, and M. Imran, “Crisismmd: Multimodal twitter datasets from natural disasters,” *Proceedings of the 12th International AAAI Conference on Web and Social Media (ICWSM)*, vol. 12, no. 1, June 2018. DOI: 10.1609/icwsml.v12i1.14983

Other relevant works

In addition to the three publications formally included in this PhD Thesis, several complementary works were produced during the doctoral period through related research activities, collaborations, projects, and parallel scientific lines.

- Conference tutorial session directly based on the PhD thesis content:
 - (*Tutorial*) Enrique Tomás Martínez Beltrán, Pedro Miguel Sánchez Sánchez, Alberto Huertas Celdrán, Gregorio Martínez Pérez, G r me Bovet, and Burkhard Stiller. (2023). *Decentralized Federated Learning: Enabling Collaborative AI With Enhanced Trust and Efficiency*. 26th European Conference on Artificial Intelligence (ECAI 2023).
- Complementary journal publications directly related to DFL and with a strong collaboration with the University of Zurich and armasuisse Science & Technology:
 - (*Journal*) Enrique Tomás Martínez Beltr n, G r me Bovet, Gregorio Mart nez P rez, and Alberto Huertas Celdr n. *Decentralized Self-Supervised Representation Learning via Prototype Exchange under Non-IID Data*. Submitted to *Future Generation Computer Systems*, 2026.
 - (*Journal*) Enrique Tomás Martínez Beltr n, Eduard Gash, G r me Bovet, Alberto Huertas Celdr n, and Burkhard Stiller. *Asynchronous Cache-based Aggregation with Fairness and Filtering for Decentralized Federated Learning*. *Computer Networks*, pp. 112225–112225, 2026. DOI: 10.1016/j.comnet.2026.112225.
 - (*Journal*) Enrique Tomás Martínez Beltr n, Philip Giryes, G r me Bovet, Burkhard Stiller, Gregorio Mart nez P rez, and Alberto Huertas Celdr n. *FedEnD: Communication-Efficient Federated Learning for Non-IID Data via Decentralized Ensemble Distillation*. *Future Generation Computer Systems*, vol. 183, pp. 108528–108528, 2026. DOI: 10.1016/j.future.2026.108528.
 - (*Journal*) Isaac Marroqui Penalva, Enrique Tomás Martínez Beltr n, Manuel Gil P rez, and Alberto Huertas Celdr n. *RepuNet: A Reputation System for Mitigating Malicious Clients in DFL*. *Computer Networks*, vol. 282, pp. 112242–112242, 2026. DOI: 10.1016/j.comnet.2026.112242.
 - (*Journal*)  ngel Luis Perales G mez, Enrique Tomás Martínez Beltr n, Pedro Miguel S nchez S nchez, and Alberto Huertas Celdr n. *TemporalFED: Detecting Cyberattacks in Industrial Time-Series Data Using Decentralized Federated*

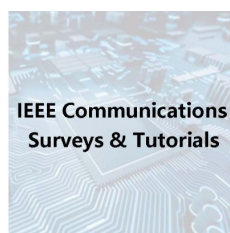
Learning. IEEE Access, vol. 14, pp. 64822–64835, 2026. DOI: 10.1109/ACCESS.2026.3681545.

- (Journal) Pedro Miguel Sánchez Sánchez, Alberto Huertas Celdrán, Enrique Tomás Martínez Beltrán, Daniel Demeter, G r me Bovet, Gregorio Mart nez P rez, and Burkhard Stiller. *Analyzing the robustness of decentralized horizontal and vertical federated learning architectures in a non-IID scenario. Applied Intelligence*, vol. 54, no. 8, pp. 6637–6653, 2024. DOI: 10.1007/s10489-024-05510-1.
- (Journal) Chao Feng, Alberto Huertas Celdr n, Jan von der Assen, Enrique Tom s Mart nez Beltr n, G r me Bovet, and Burkhard Stiller. *DART: A Solution for decentralized federated learning model robustness analysis. Array*, vol. 23, pp. 100360–100360, 2024. DOI: 10.1016/j.array.2024.100360.
- (Journal) Jan von der Assen, Sandrin Raphael Hunkeler, Alberto Huertas Celdr n, Enrique Tom s Mart nez Beltr n, G r me Bovet, and Burkhard Stiller. *Reputation System based on Distributed Ledger to Secure Decentralized Federated Learning*. Research Square, submitted to *International Journal of Information Security*, 2024. DOI: 10.21203/rs.3.rs-4997851/v1.
- National Spanish cybersecurity conference contributions, mainly within the Jornadas Nacionales de Investigaci n en Ciberseguridad (JNIC):
 - (Conference) Pedro Beltr n L pez, Enrique Tom s Mart nez Beltr n, Pantaleone Nespoli, Manuel Gil P rez, Gregorio Mart nez P rez, and Alberto Huertas Celdr n. *MadHoney: Se uelos T xicos para la Defensa Activa en el Aprendizaje Federado Descentralizado*. XI Jornadas Nacionales de Investigaci n en Ciberseguridad, JNIC 2026, Barcelona, Spain, 2026.
 - (Conference) Enrique Tom s Mart nez Beltr n, Miguel Fern ndez Llamas, Anas Zine Boujemaoui, G r me Bovet, Gregorio Mart nez P rez, and Alberto Huertas Celdr n. *Aprendizaje federado descentralizado y conciencia situacional para el reconocimiento a reo militar seguro y resiliente*. X Jornadas Nacionales de Investigaci n en Ciberseguridad, JNIC 2025, Zaragoza, Spain, 2025. *This contribution is documented as receiving the award for best original research*.
 - (Conference) Enrique Tom s Mart nez Beltr n, Pedro Miguel S nchez S nchez, G r me Bovet, Gregorio Mart nez P rez, and Alberto Huertas Celdr n. *Mitigaci n de ataques bizantinos usando modelos hist ricos en aprendizaje federado descentralizado*. IX Jornadas Nacionales de Investigaci n en Ciberseguridad, JNIC 2024, Sevilla, Spain, 2024.
 - (Conference) Enrique Tom s Mart nez Beltr n, Mario Quiles P rez, Sergio L pez Bernal, Alberto Huertas Celdr n, and Gregorio Mart nez P rez. *A Review of Noise-based Cyberattacks Generating Fake P300 Waves in Brain-Computer Interfaces*. VII Jornadas Nacionales de Investigaci n en Ciberseguridad, JNIC 2022, Bilbao, Spain, 2022.
- International conference contributions related to DFL, communication efficiency, adversarial robustness, and military or cybersecurity scenarios:
 - (Conference) Pedro Miguel S nchez S nchez, Enrique Tom s Mart nez Beltr n, Chao Feng, G r me Bovet, Gregorio Mart nez P rez, and Alberto Huertas Celdr n. *S-VOTE: Similarity-based Voting for Client Selection in Decentralized*

- Federated Learning*. International Joint Conference on Neural Networks, IJCNN 2025, Rome, Italy, pp. 1–8, 2025. DOI: 10.1109/IJCNN64981.2025.11227231.
- (Conference) Pedro Miguel Sánchez Sánchez, Enrique Tomás Martínez Beltrán, Miguel Fernández Llamas, G r me Bovet, Gregorio Mart nez P rez, and Alberto Huertas Celdr n. *ProFe: Communication-Efficient Decentralized Federated Learning via Distillation and Prototypes*. IEEE International Conference on Communications, ICC 2025, Montreal, Canada, pp. 1596–1601, 2025. DOI: 10.1109/ICC52391.2025.11161228.
 - (Conference) Chao Feng, Alberto Huertas Celdr n, Janosch Baltensperger, Enrique Tom s Mart nez Beltr n, Pedro Miguel S nchez S nchez, G r me Bovet, and Burkhard Stiller. *Sentinel: An Aggregation Function to Secure Decentralized Federated Learning*. 27th European Conference on Artificial Intelligence, ECAI 2024, Santiago de Compostela, Spain, pp. 1760–1767, 2024. DOI: 10.3233/FAIA240686.
 - (Conference) Pedro Miguel S nchez S nchez, Enrique Tom s Mart nez Beltr n, Alberto Huertas Celdr n, Robin Wassink, G r me Bovet, Gregorio Mart nez P rez, and Burkhard Stiller. *Stealth Spectrum Sensing Data Falsification Attacks Affecting IoT Spectrum Monitors on the Battlefield*. IEEE Military Communications Conference, MILCOM 2023, Boston, USA, pp. 673–678, 2023. DOI: 10.1109/MILCOM58377.2023.10356249.

Publications composing
the PhD Thesis

Decentralized Federated Learning: Fundamentals, State of the Art, Frameworks, Trends, and Challenges



Title:	Decentralized Federated Learning: Fundamentals, State of the Art, Frameworks, Trends, and Challenges
Authors:	Enrique Tomás Martínez Beltrán, Mario Quiles Pérez, Pedro Miguel Sánchez Sánchez, Sergio López Bernal, G�r�me Bovet, Manuel Gil P�rez, Gregorio Mart�nez P�rez, Alberto Huertas Celdr�n.
Journal:	IEEE Communications Surveys and Tutorials
JIF:	34.4 Q1-D1 (2023)
Publisher:	IEEE
Volume:	25
Number:	4
Pages:	2983–3013
Month:	Nov
Year:	2023
DOI:	10.1109/COMST.2023.3315746
Status:	Published

Abstract

In recent years, Federated Learning (FL) has gained relevance in training collaborative models without sharing sensitive data. Since its birth, Centralized FL (CFL) has been the most common approach in the literature, where a central entity creates a global model. However, a centralized approach leads to increased latency due to bottlenecks, heightened vulnerability to system failures, and trustworthiness concerns affecting the entity responsible for the global model creation. Decentralized Federated Learning (DFL) emerged to address these concerns by promoting decentralized model aggregation and minimizing reliance on centralized architectures. However, despite the work done in DFL, the literature has not (i) studied the main aspects differentiating DFL and CFL; (ii) analyzed DFL frameworks to create and evaluate new solutions; and (iii) reviewed application scenarios using DFL. Thus, this article identifies and analyzes the main fundamentals of DFL in terms of federation architectures, topologies, communication mechanisms, security approaches, and key performance indicators. Additionally, the paper at hand explores existing mechanisms to optimize critical DFL fundamentals. Then, the most relevant features of the current DFL frameworks are reviewed and compared. After that, it analyzes the most used DFL

application scenarios, identifying solutions based on the fundamentals and frameworks previously defined. Finally, the evolution of existing DFL solutions is studied to provide a list of trends, lessons learned, and open challenges.

Keywords

Decentralized Federated Learning · Communication Mechanisms · Security and Privacy
· Key Performance Indicators · Frameworks · Application Scenarios

Decentralized Federated Learning: Fundamentals, State of the Art, Frameworks, Trends, and Challenges

Enrique Tomás Martínez Beltrán^{1b}, Mario Quiles Pérez,
Pedro Miguel Sánchez Sánchez^{1b}, *Student Member, IEEE*, Sergio López Bernal^{1b}, G  r  me Bovet^{1b},
Manuel Gil P  rez^{1b}, Gregorio Mart  nez P  rez^{1b}, and Alberto Huertas Celdr  n^{1b}, *Member, IEEE*

Abstract—In recent years, Federated Learning (FL) has gained relevance in training collaborative models without sharing sensitive data. Since its birth, Centralized FL (CFL) has been the most common approach in the literature, where a central entity creates a global model. However, a centralized approach leads to increased latency due to bottlenecks, heightened vulnerability to system failures, and trustworthiness concerns affecting the entity responsible for the global model creation. Decentralized Federated Learning (DFL) emerged to address these concerns by promoting decentralized model aggregation and minimizing reliance on centralized architectures. However, despite the work done in DFL, the literature has not (i) studied the main aspects differentiating DFL and CFL; (ii) analyzed DFL frameworks to create and evaluate new solutions; and (iii) reviewed application scenarios using DFL. Thus, this article identifies and analyzes the main fundamentals of DFL in terms of federation architectures, topologies, communication mechanisms, security approaches, and key performance indicators. Additionally, the paper at hand explores existing mechanisms to optimize critical DFL fundamentals. Then, the most relevant features of the current DFL frameworks are reviewed and compared. After that, it analyzes the most used DFL application scenarios, identifying solutions based on the fundamentals and frameworks previously defined. Finally, the evolution of existing DFL solutions is studied to provide a list of trends, lessons learned, and open challenges.

Index Terms—Decentralized federated learning, communication mechanisms, security and privacy, key performance indicators, frameworks, application scenarios.

I. INTRODUCTION

ARTIFICIAL Intelligence (AI) and, in particular, Machine Learning (ML), will be one of the techniques that benefit from the vast amount of data expected in the coming years. However, data originating from a massive number of Internet of Things (IoT) devices are commonly stored in a distributed manner for a wide range of scenarios such as smart grids [1], remote health monitoring [2], or Internet of Vehicles (IoV) [3]. In such cases, data collection in central entities, as traditionally done in ML, is often infeasible or impractical due to limited communication resources, data privacy concerns, or country regulations. In 2016, Federated Learning (FL) [4] solved this issue by allowing entities (also known as participants, clients, or nodes of a federation) to train collaborative models without sharing training data. FL can be classified into two categories according to how federated models are created: centralized, known as Centralized Federated Learning (CFL), and decentralized, known as Decentralized Federated Learning (DFL). Nowadays, CFL is the predominant FL approach. It considers a central orchestration server to create and distribute a global model to the rest of the participants or clients. More in detail, clients train their models with local data, then send the local model parameters to a central server, where a global model is created by aggregating and combining the parameters of the individual model. Figure 1 shows the timeline since Google introduced FL and how new features and approaches for decentralized scenarios have emerged. As can be seen, the growth for CFL and DFL is remarkable.

DFL, also known as Serverless or Distributed FL, emerged in 2018 to distribute the aggregation of model parameters between the neighboring participants [5]. The operation of DFL focuses on transmitting fast updates computed locally by each node (e.g., model parameters or gradients) and meta-data (e.g., activations functions in neural networks) to the rest of the federation nodes. Therefore, compared to CFL, DFL improves the limitations of having a single point of failure, trust dependencies, and bottlenecks at the server node [6].

Manuscript received 24 April 2023; revised 11 July 2023 and 14 August 2023; accepted 4 September 2023. Date of publication 15 September 2023; date of current version 22 November 2023. This work was supported in part by the Fundaci  n S  neca, Regi  n de Murcia, Spain, under Grant 21629/FPI/21; in part by MCIN/AEI/10.13039/501100011033, NextGenerationEU/PRTR, UE, under Grant TED2021-129300B-I00; in part by MCIN/AEI/10.13039/501100011033/FEDER, UE, under Grant PID2021-122466OB-I00; in part by the Swiss Federal Office for Defense Procurement (armasuisse) with the DEFENDIS and CyberForce Projects; and in part by the University of Z  rich (UZH). (*Corresponding author: Enrique Tom  s Mart  nez Beltr  n.*)

Enrique Tom  s Mart  nez Beltr  n, Mario Quiles P  rez, Pedro Miguel S  nchez S  nchez, Sergio L  pez Bernal, Manuel Gil P  rez, and Gregorio Mart  nez P  rez are with the Department of Information and Communications Engineering, University of Murcia, 30100 Murcia, Spain (e-mail: enriquetomas@um.es; mqp@um.es; pedromiguel.sanchez@um.es; slopez@um.es; mgilperez@um.es; gregorio@um.es).

G  r  me Bovet is with the Cyber-Defence Campus, Armasuisse Science and Technology, 3602 Thun, Switzerland (e-mail: gerome.bovet@armasuisse.ch).

Alberto Huertas Celdr  n is with the Communication Systems Group, Department of Informatics, University of Zurich, 8050 Z  rich, Switzerland (e-mail: huertas@ifi.uzh.ch).

Digital Object Identifier 10.1109/COMST.2023.3315746

1553-877X    2023 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.
See <https://www.ieee.org/publications/rights/index.html> for more information.

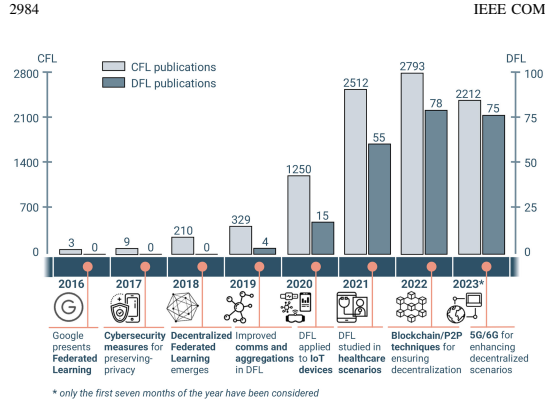


Fig. 1. DFL Timeline (source: Web of Science).

Firstly, DFL improves fault tolerance because nodes constantly update their knowledge about available nodes or those that have ceased communicating [7]. This enhances the robustness of the network and mitigates the risk of a single point of failure, as it happens in CFL. Secondly, DFL improves trust issues by distributing trust between the federation nodes, which determines the trust of the entire federation, drastically reducing the single point of attack. Thirdly, DFL reduces the network bottleneck issue by allowing more evenly distributed communication and workload among the nodes, thus reducing the chances of congestion or delays in the overall network performance. Furthermore, DFL provides self-scaling federations, as new nodes can join the network and start communicating with other nodes, considering factors such as device throughput or computational capacity. Although the number of connections may decrease due to the limited number of nodes each node can share data with, the network remains tightly connected [4]. Finally, DFL enables more efficient use of resources by distributing the computing power required to aggregate the model parameters among all participating nodes, rather than relying on a single parameter server under centralized control [8].

Despite the benefits provided by DFL, it also introduces new challenges in areas such as communication overhead, training optimization, and trustworthy AI. Depending on how the model aggregation is distributed across the network, some DFL topologies may face increased communication overhead [9]. In these cases, careful design and optimization of the communication protocols, client selection strategies, and trust mechanisms can help address these limitations. Trust plays a critical role in these tasks, as it influences the decision of which clients to share and aggregate model parameters with. Additionally, the significant amount of model parameters exchanged from the network edge to the data centers carries the risk of saturating the backbone. As a result, using DFL necessitates reevaluating the infrastructure from a new perspective, as it reduces the need for centralized infrastructure while aiming to improve network performance and trustworthiness. In this context, trust becomes essential in addressing the emerging challenges of DFL, such as client

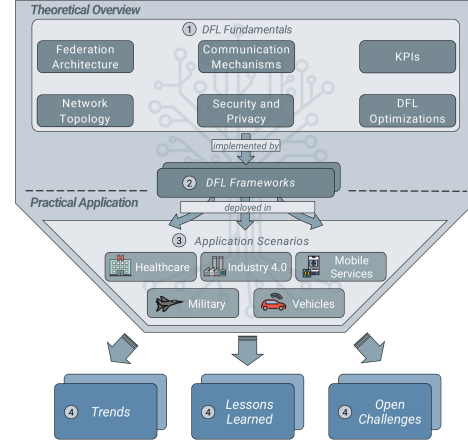


Fig. 2. Methodological steps and storyline of this work.

selection and parameter sharing. These challenges warrant further exploration and research, as the paper at hand does.

To explore the research field of DFL in an organized fashion, the paper follows the methodological steps indicated in Figure 2. First, it is essential to identify and study the key aspects that differentiate DFL from CFL (step 1 in Figure 2). For this purpose, the literature has highlighted some elements such as (i) federation architecture, in charge of modeling the decentralized scenario based on participants, roles, decentralization of the nodes, and distribution of data features; (ii) network topology, defining the associations between the nodes of the federation [10]; (iii) communication mechanisms, orchestrating the exchange of model parameters within the federation [11]; and (iv) security and privacy, studying possible cyberattacks and the countermeasures to preserve data privacy and models robustness [12]. In addition to the above fundamentals, identifying Key Performance Indicators (KPIs) is important for assessing DFL performance [13]. Three perspectives can be explored in this context: nodes, communications, and models. The first focuses on evaluating the heterogeneity and dynamism of nodes in DFL, the second on the efficiency of inter-node communications for data exchange, and the third on the performance of ML/DL models for solving collaborative tasks. Finally, mechanisms to optimize KPIs are also essential to ensure the efficiency and multi-scenario adaptability of the approach [14]. In this context, this article expands upon the existing literature by comprehensively identifying and describing the DFL fundamentals, KPIs, and optimization mechanisms.

The aforementioned fundamentals are crucial for developing and comparing frameworks capable of training DFL-based models (step 2 in Figure 2). These fundamentals are particularly important because they guide frameworks on managing model parameter exchanges efficiently, establishing federations with network participants, implementing effective aggregation mechanisms for local models, and fostering more reliable and trustworthy models. By

considering these fundamentals, DFL frameworks can ensure better performance, security, scalability, and trustworthiness across a wide range of use cases [15]. Therefore, the article at hand also aims to support researchers working in decentralized systems using federated approaches with a review of the most relevant features of existing frameworks.

Then, application scenarios of DFL should be analyzed according to the fundamentals and frameworks they implement (step 3 in Figure 2). At this point, it is important to identify strengths and weaknesses, and enables more informed decisions for selecting and deploying suitable approaches in real-world scenarios. Nowadays, the most used scenarios are: (i) healthcare, favoring the decentralization of clinical records and collaborative diagnosis [16]; (ii) Industry 4.0, improving the efficiency of automated industrial systems [17]; (iii) mobile services, decreasing response times and increasing the bandwidth of constraints devices [18]; (iv) military, creating robust networks between Unmanned Aerial Vehicles (UAVs) and protecting them with cyber defense mechanisms [19]; and (v) vehicles, ensuring high mobility and local storage management [20]. Subsequently, each scenario is explored by studying the proposed solutions according to the previous fundamentals. Finally, all the aspects highlighted above (fundamentals, frameworks, and application scenarios) are analyzed with a view into the past, present, and future to extract trends, lessons learned, and challenges in the field of DFL. This is the last step of the followed methodology (step 4 in Figure 2), which also shares the same structure and storyline as the document at hand.

II. MOTIVATION AND CONTRIBUTIONS

DFL is an encouraging field of research that has repeatedly been reported as challenging in several review articles in recent years. While there are numerous surveys on FL, most focus primarily on the CFL-based approach and provide limited coverage of DFL. Therefore, to the best of our knowledge, this is the first survey providing a comprehensive literature review of DFL. The increase in the number of papers using DFL in different scenarios: healthcare [2], [21], Industry 4.0 [17], mobile services [18], military [19], and vehicles [22] motivate the need to identify the fundamental elements of DFL and review the work done to date.

Table I compares the most relevant surveys analyzing both CFL and DFL approaches. Regarding CFL, several works have attracted immense interest from academia and industry [37]. In this sense, in 2020, Lim et al. [23] reviewed the use of CFL in mobile networks using different end-user applications, while in 2021, researchers such as Nguyen et al. [24] and Khan et al. [25] examined the application of FL in more complex scenarios involving different heterogeneous IoT devices. Regarding security and privacy, Mothukuri et al. [26] defined secure protocols between server and participants, while Boobalan et al. [27] provided solid CFL fundamentals. Other articles, such as [28], reviewed the applicability in specific medical scenarios, discussing sensitive data privacy mechanisms. Finally, Witt et al. [29] detailed the frameworks most commonly used.

In contrast, most of the works covering DFL approaches focus on analyzing the use of Distributed Ledger Technology (DLT) technologies (e.g., Blockchain) [30]. Similarly, other authors explored the applicability of DFL in particular scenarios such as IoV [31], wireless communications [32], or UAV devices [33]. Moreover, the article by Wu et al. [34] provided a comprehensive survey of optimized DFL models and algorithms focusing on network topologies. The authors introduced various topologies with different participants and potential optimizations for them. Additionally, Chen et al. [35] analyzed the advances in FL, including aspects such as communications, privacy, and security requirements in DFL. Also, they provided solutions for collaborative training using more generalized algorithms. Further research has also been conducted on the frameworks that enable DFL. Witt et al. [36] proposed a set of solutions that address privacy and security issues in DFL. The authors also provided reward systems based on smart contracts to incentivize honest customer participation in the federated process. These latter works address the performance of DFL in selected application scenarios and detail critical elements of superiority over CFL approaches.

Despite the contributions of previous works, as illustrated in Table I, none addresses a literature review of DFL. In addition, no previous research studies the fundamentals of the emerging DFL approach. Similarly, recent work does not review DFL solutions from a fundamentals perspective as elements of analysis and comparison for different application scenarios. These solutions are often deployed on frameworks capable of managing the proposed federated approach [31], [38]. However, no study highlights which frameworks are feasible for this new decentralized approach nor collects the latest developments. Within this context, and adhering to the storyline outlined in Section I, it becomes essential to address the following research questions:

- *Q1. What are the fundamental aspects of DFL?* Depending on the federation architecture of the nodes, network topology, communication mechanisms between nodes, or security techniques, variations in the DFL approach may occur. Similarly, approach-based performance metrics and optimization techniques are needed. However, solutions in the literature ignore details about these elements and how they are combined.
- *Q2. What DFL frameworks exist, and what fundamentals do they provide?* There is no study reviewing existing frameworks able to build DFL solutions, analyzing their characteristics, and defining which application scenarios can be utilized.
- *Q3. Which are the main characteristics of the most relevant scenarios of DFL?* It is necessary to analyze and compare the previous DFL fundamentals and frameworks to solve the problems motivated by each application scenario. In addition, it is also necessary to detect the characteristics and limitations posed by existing solutions. The literature has not studied these approaches from a broad perspective to have a complete view.
- *Q4. What trends, lessons learned, and challenges have emerged in DFL?* To establish the guidelines for future research, it is critical to describe how DFL has evolved

TABLE I
COMPARISON OF SURVEYS ANALYZING CFL AND DFL

Ref.	Year	FL Approach	Device Types / Area	Fundamentals	Frameworks	Application Scenarios	Key Performance Indicators	Focus and Solution Categorization
[23]	2020	CFL	Mobile networks	✓	✓	✗	✓	- A survey on the integration of FL and edge computing. - The applications of FL in IoT networks and services have not been explored and discussed.
[24]	2021	CFL	IoT devices	✗	✗	✓	✗	- A survey on FL in an IoT scenario, reviewing data distribution, ML models, privacy mechanism, and communication architecture. - The study has not discussed other approaches or frameworks capable of managing the scenario.
[25]	2021	CFL	IoT devices	✓	✗	✓	✓	- A systematic review of FL in IoT scenarios, detailing the application areas and limitations. - The study only considers CFL architectures without discussing the disadvantages compared to other approaches.
[26]	2021	CFL	Security Privacy	✗	✓	✓	✗	- A survey of FL security and privacy, defining secure protocols. - The analysis of secure measures using different FL approaches is missing.
[27]	2022	CFL	FL baselines	✓	✓	?	?	- A survey on the FL baselines with a basic introduction to definitions and architectures. - The application scenarios of FL have not been discussed.
[28]	2022	CFL	Healthcare	?	✗	✓	✗	- An overview of the use of FL in Healthcare scenarios. - Limited overview of techniques for deploying realistic scenarios.
[29]	2022	CFL	Framework review	✓	✓	✗	✗	- An overview of frameworks for deploying DFL-based architectures. - Lack of identification of application scenarios.
[30]	2022	DFL	DLT	✗	✗	✗	?	- A description of the challenges and applications of Blockchain. - The paper only focuses on the Blockchain technology to ensure node decentralization.
[31]	2022	DFL	IoV	✗	✓	✓	?	- A brief description of FL in fog radio access networks. - The applications of FL in IoT networks have not been presented.
[32]	2022	DFL	Wireless communications	✗	✗	✓	✓	- A review of the techniques for adapting FL to distributed environments. - The paper only reviews techniques without giving specific application scenarios or frameworks.
[33]	2022	DFL	UAV devices	✗	✗	✓	✓	- A brief survey on the application of FL in UAV networks. - Other domains, such as smart city or IoT devices, are not considered.
[34]	2023	DFL	DFL baselines	?	✗	?	✓	- A study of optimized DFL models and algorithms focusing on network topologies. - Limited review regarding federation architectures, device heterogeneity, and frameworks.
[35]	2023	DFL	DFL baselines	?	✗	✓	✓	- A comparison of CFL and DFL federation architectures regarding topologies, privacy, and security. - The study does not have enough information about DFL, including algorithms, optimizations, or metrics for evaluating its performance.
[36]	2023	DFL	Framework review	?	✓	✗	✗	- A systematic review of DFL frameworks, highlighting differences, limitations, and future research directions. - The study focuses only on Blockchain-related frameworks and participant reward techniques.
This work	2023	DFL	Healthcare Industry 4.0 Mobile services Military Vehicles	✓	✓	✓	✓	- A survey on the baselines and fundamentals of DFL, generating a novel taxonomy in the literature. - The paper explores the application of DFL solutions and frameworks in various areas of interest.

✓ fully addressed, ? partially addressed, ✗ not addressed by the work

over the last years and what trends and open challenges are in the field.

To answer the previous research questions and provide readers with an up-to-date vision of DFL, the main contributions of this manuscript are:

- An analysis of the fundamentals that compose DFL: (i) federation architecture, (ii) network topology, (iii) communication mechanisms, (iv) security and privacy, (v) KPIs, and (vi) techniques to optimize the above KPIs, paying attention to the studies and applications that address each fundamental (answering *Q1* in Section III).
- A description of the main open-source frameworks to create and manage DFL solutions. This description is divided into mature literature solutions that have been redesigned to offer decentralized architectures and incipient solutions that address specific scenarios (answering *Q2* in Section IV).
- A comprehensive review and comparison of the characteristics, advantages, and limitations of the most relevant solutions in the literature on the most used application scenarios: (i) healthcare, (ii) Industry 4.0, (iii) mobile services, (iv) military, and (v) vehicles (answering *Q3* in Section V).

- A set of current trends, lessons learned, and future challenges drawn from DFL works and frameworks reviewed (answering *Q4* in Section VI).

Figure 3 shows where and how the above questions are addressed in the paper at hand, acting as a table of contents. In particular, Section III analyzes the fundamentals of DFL in terms of architectures, topologies, communications, security, performance indicators, and techniques for optimizing scenarios. Section IV examines the main open-source frameworks that support the deployment of DFL with the previously defined fundamentals. After that, Section V describes and compares the leading solutions found in the state of the art, analyzing deployment considerations, the robustness of the proposed solution, and the results obtained. Section VI draws a set of lessons learned, current trends, and future challenges in the research area. Finally, Section VII provides an insight into the conclusions extracted from the work.

III. FUNDAMENTALS AND TAXONOMY

The fundamentals of DFL establish the groundwork for analyzing existing frameworks and solutions in the literature. Following the storyline presented in Figure 2, this

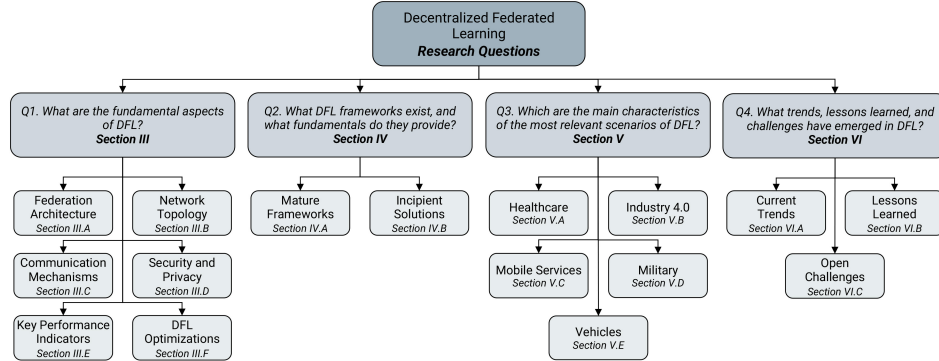


Fig. 3. Research questions and sections answering them.

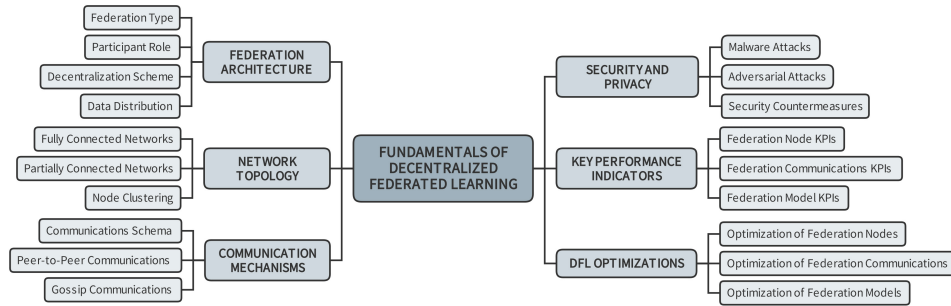


Fig. 4. Taxonomy with the fundamental aspects of DFL.

section responds to “Q1. What are the fundamental aspects of DFL?,” identifying and analyzing the following fundamentals: (i) federation architectures, (ii) network topology, (iii) communication mechanisms, (iv) security and privacy, (v) KPIs, and (vi) techniques to optimize the previous metrics. These fundamentals are detailed in Figure 4, showing the subcategories that compose them and the resulting taxonomy. Additionally, Table II details the above taxonomy with a detailed definition of the various fundamentals and the literature solutions that address them.

A. Federation Architecture

DFL solutions require an architecture that allows effective collaboration and communication between network participants. In this sense, this work analyzes the fundamentals based on the federation type in the communication, differentiating organizations or data centers from other devices. Subsequently, DFL architectures can also be categorized based on the roles of the participants. In this classification, each participant is assigned a specific task within the federated learning process. Furthermore, the decentralization schema differentiates between a decentralized, semi-decentralized, and centralized architecture. The semi-decentralized approach can be seen as a hybrid between centralized and decentralized architectures, offering a balance of both. Finally, the definition of DFL architectures is based on the data distribution during federation.

In the following sections, each category will be discussed in greater detail.

1) *Federation Type*: The literature categorizes DFL architectures according to the type of federation, which is based on the participating nodes: cross-silo and cross-device. The differences lie in the number of participants and the amount of data stored in each one [39].

In cross-silo DFL, the nodes are organizations or data centers [40]. There is usually a relatively small number of nodes (<100), each with a large amount of data (about millions of samples), perhaps distributed and aggregated from consumers in different businesses. At the same time, nodes use consistent, robust, and scalable computing over time. Likewise, these nodes have a high performance in the network, avoiding points of failure during communications between nodes. For example, it is applied in different hospitals by training a federated model for tumor classification while keeping their Positron Emission Tomography (PET) images locally [116].

In cross-device DFL, the number of nodes is relatively large (>100), where each node has a relatively small amount of data (about thousands of samples) and limited computational power [41]. Due to concerns regarding power consumption, individual devices cannot be asked to perform complex training tasks. Furthermore, nodes could periodically disconnect from the network so that the network dropout rate would increase considerably, negatively impacting DFL.

TABLE II
COMPARISON OF SOLUTIONS ADDRESSING DIFFERENT DFL FUNDAMENTALS

Taxonomy	Fundamentals	Definition and Solutions Analyzed in the Literature	References
Federation Architecture	Federation Type	• Federation from data silos or data centers • Federation from different types of devices, such as mobile phones or IoT devices	[39], [39], [40] [2], [41], [42]
	Participant Role	• Identify the functionality of participants during federation	[43], [44]
	Decentralization Schema	• Techniques for enabling DFL while maintaining model convergence and data privacy • Balance the power and leadership in SDFL while maintaining the benefits of DFL • Combine centralized and decentralized elements to optimize FL performance and scalability	[6], [45], [46] [14], [47] [48]
	Data Distribution	• Solutions for HFL, VFL, and TFL partitioning the dataset • Explore ways to sample data to ensure better distribution across nodes	[42], [49] [50]
Network Topology	Fully Connected Networks	• Define the federated networks with participants connected to each other • Measure of robustness, flexibility, fault tolerance, cost, or security	[9], [51] [10], [52]
	Partially Connected Networks	• Define the classification in star, ring, or random structured networks	[53], [54]
	Node Clustering	• Measure of robustness, flexibility, fault tolerance, cost or security • Define and classify the federated networks based on participants clusters	[55], [56] [44], [57]
		• Measure of robustness, flexibility, fault tolerance, cost or security	[58]
Communication Mechanisms	Communications Scheme	• Data exchange where all participants transmit their models at the same time • Update participants models independently at different times	[11], [59] [60], [61]
	Peer-to-Peer Communications	• Methods for designing P2P topologies in DFL • Develop techniques for disseminating data and models in P2P networks	[62]–[64] [65]–[68]
	Gossip Communications	• Methods to design gossip protocols in DFL • Strategies to select nodes and optimize the efficiency in gossip-based communication	[69], [70] [71]–[74]
		• Determine the impact of malware threats in decentralized scenarios	[75]–[77]
Security and Privacy	Malware Attacks	• Identify the risks associated with fully connected, star-structured, and random networks	[12], [49], [56]
	Adversarial Attacks	• Identify adversarial attacks affecting decentralized architectures during collaborative federation • Compare the adversarial attacks between different FL approaches and their adaptive capacity	[49], [78], [79] [80], [81]
	Security Countermeasures	• Secure techniques to protect the transmission of model parameters between participants • Encryption mechanisms while preserving reliability between participants	[4], [82]–[84] [45], [83], [85]
		• Measure the network performance considering the heterogeneity and dynamism of the network • Quantify the resource capabilities and node mobility of participating nodes in DFL scenarios	[13], [86] [7], [87]
Key Performance Indicators	Federation Nodes KPIs	• Measure the reliability of model parameter exchanges between participants in DFL • Evaluate communications flexibility and overhead in DFL	[10], [88] [7], [89]
	Federation Communications KPIs	• Measure the performance by solving different tasks or using multiple datasets for benchmarking • Mechanisms to ensure the trustworthiness of federated models	[13], [90], [91] [92]–[96]
	Federation Models KPIs	• Optimal methods for selecting nodes to transmit model parameters • Optimize algorithms to suit the heterogeneous node resources and capabilities	[97], [98] [26], [58], [99]
		• Efficient distribution schemes that minimize the number of network exchanges • Compression techniques to reduce the amount of data that needs to be exchanged	[71], [100]–[102] [103]–[107]
DFL Optimizations	Optimization of Federation Nodes	• Explore efficient ML models to increase model performance and reliability during federation • Use of meta-learning, multitasking, and federated distillation in federation tasks	[88], [108]–[111] [91], [112]–[115]
	Optimization of Federation Communications		

performance. The nodes are usually on-edge devices or robots like UAVs [42], where communications between devices are often weak if not kept in a close coverage radius.

2) *Participant Role*: Network nodes may have one or more roles that define their behavior and operation in DFL architectures. In particular, the existing roles are trainer, aggregator, proxy, and idle. A trainer node aims to train a local model with its local dataset and transmit the parameters to its neighboring nodes [43]. The trainer node expects to receive the parameters of the updated federated model to incorporate it back into the local model. In contrast, the node with the aggregator role is responsible for obtaining the parameters from the neighboring nodes, aggregating them in the global model, and transmitting them to the neighboring nodes. In specific network topologies (see Section III-B), aggregators are not directly reachable by other nodes, and proxy nodes are needed. This role is intended to relay the received model parameters to neighboring nodes, allowing different nodes or network topologies to be interconnected [44]. However, a participant may not have any of the above roles, being idle in the network and not participating in the federation.

3) *Decentralization Schema*: The federation architecture can be affected by the decentralization level maintained between participants, with three different approaches emerging (see Figure 5): DFL, Semi-Decentralized Federated Learning (SDFL), and CFL. In DFL, participants perform four steps independently: local model training, parameter exchange, local model aggregation, and parameter exchange again. In SDFL, participants perform the first two steps, while an aggregator participant handles the third step and transfers leadership for

the aggregation functionality (step 5). In CFL, a central server handles parameter aggregation (step 3), with the rest of the network receiving and updating their local models accordingly (steps 4 and 5).

In DFL architectures, the network is fully autonomous, managing communications with other nodes and aggregating the model parameters independently of the other nodes. Each network node can select one or more nodes to transmit data, defining bidirectional communications between them. Furthermore, decentralization can be assumed to be fixed or dynamic, as the interconnections between nodes can change over time. Unlike the node-server architecture, the absence of a centralized server in DFL allows for the relaxation of the synchronous model update. Different solutions in the literature make several assumptions about network connectivity, particularly considering that each node is connected to all other nodes in the network or only to a set of neighboring nodes. Therefore, issues such as fixed or dynamic topology arise and the use of directed or undirected graphs [46]. A fixed topology provides stability and predictability but limits adaptability, while a dynamic topology allows flexibility but requires additional overhead to maintain connectivity. In contrast, directed graphs model asymmetric relationships, and undirected graphs are more flexible for modeling symmetric relationships between participants.

In contrast to the previous approach, SDFL architectures maintain an aggregator role that rotates among the participants belonging to the network. The aggregator transmits the leadership (aggregation functionality) periodically during the federation, selecting the neighboring node randomly or

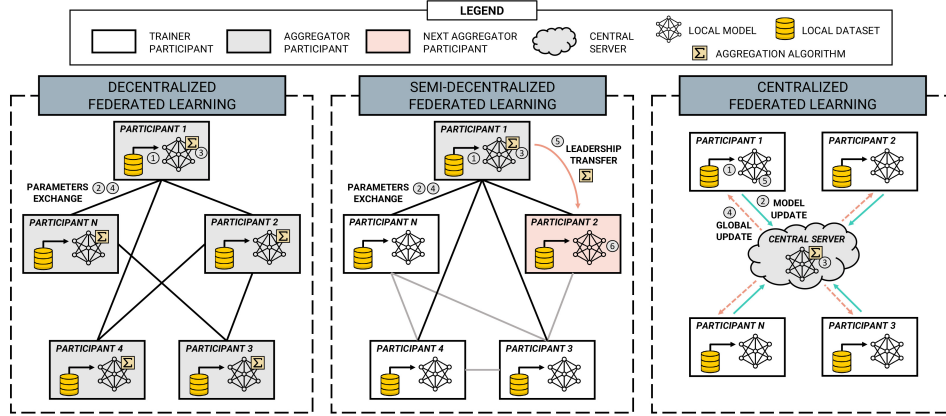


Fig. 5. Common approaches based on DFL architecture decentralization. The numbers inside the circles represent the training lifecycle.

based on its performance, such as network, computational, or power capacity [47], [48]. The leadership in SDFL architectures plays an essential role in determining which node is responsible for collecting the model updates and computing the next-generation model by aggregation. Also, the choice of leadership selection mechanism can significantly impact the performance, efficiency, and robustness of the system.

Finally, in CFL-based architectures, data decentralization is orchestrated by a single aggregator. This central node aggregates the model parameters shared periodically by each trainer and creates a robust central model distributed among the nodes [6]. In this approach, the aggregating participant is fixed during federation, with the central node being responsible for aggregating model parameters for each trainer in the network. Additionally, the CFL architecture makes it possible to ignore problems related to topology and device performance by not having decentralized aggregators. To improve this situation, techniques such as Multiparty Computation (MPC) can guarantee the immutability of model updates and coordination between participants directly, as could be the case for healthcare institutions [45].

4) *Data Distribution:* The DFL approach, like CFL, has different configurations depending on the properties of the distributed data. Data can be Independent and Identically Distributed (IID) when the federation participants behave similarly. Therefore, the data distribution does not fluctuate ($x^i \sim D$), and the events associated with the data points are independent, i.e., they are not connected to each other in any way ($\nexists j \ p(x^i, x^j) = p(x^i) \ p(x^j)$), where p represents the joint probability distribution of the data points. However, in some application scenarios, nodes are usually heterogeneous. Hence, the data often have a non-IID format, where data vary in quality, diversity, and quantity in the network, thus increasing the complexity of modeling, analysis, and evaluation. The problem of heterogeneous data distribution poses a challenge for DFL. It can cause local model parameters to converge at different stationary points for different participants, making global convergence difficult. In addition, there might be

multiple aggregated models during the federation where nodes are connected irregularly, leading to different global models receiving data with varying distributions. Addressing this challenge involves determining how to handle these various models. One option presented in the literature is to continue aggregating them at a higher level [117], while another option is to accept the presence of different federated models based on the network topology and characteristics of the nodes [118]. By considering these alternatives, DFL can better address the issues arising from non-IID data distributions.

Finally, depending on the nature of the problem to be solved and the organization of data (features and samples) among nodes, DFL is usually divided into three types: Horizontal Federated Learning (HFL) [42], Vertical Federated Learning (VFL) [49], and Transfer Federated Learning (TFL) [50]. HFL is the most commonly employed method in DFL, as it involves sample federation and is applicable when there are many overlapping features and few overlapping nodes, typically associated with cross-device scenarios. In contrast, VFL and TFL require more in-depth analysis and comparison, as they are more complex to adapt and deploy in application scenarios that take advantage of their unique data organization. VFL focuses on feature binding when there are many overlapping nodes and few overlapping features, while TFL is considered when there is a limited feature and sample intersection between nodes. The goal of TFL is to build efficient models for specific applications in cases where data are sparse, which makes it another relevant aspect of DFL strategies.

The successful transition from CFL to DFL involves the adaptation of commonly used CFL datasets. Techniques like decentralized sparse partitioning prove instrumental in aligning these datasets with the unique characteristics of DFL [38]. These techniques consider the aggregation algorithms, participants' roles, and the task to be solved. Similarly, the distinct data distribution across nodes in DFL could be addressed using federated sampling techniques, which help maintain data diversity while ensuring statistical accuracy [119]. These adaptations make it feasible to use CFL datasets in DFL

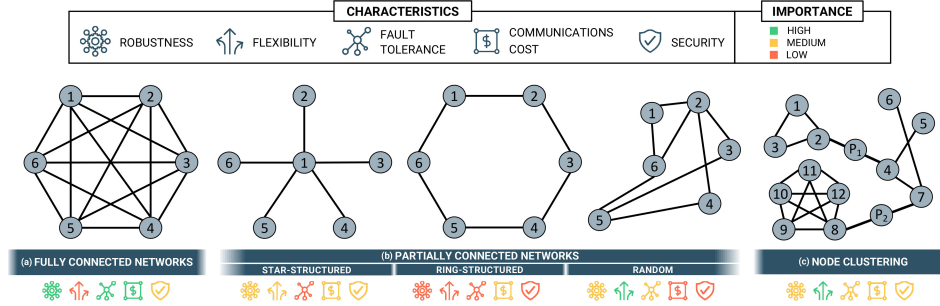


Fig. 6. DFL network topologies. (a) Fully connected networks, (b) Partially connected networks (star-structured network, ring-structured network, random network), (c) Node clustering. P_n indicates the node acting as a proxy.

scenarios efficiently, leading to optimized learning processes and improved performance of DFL implementations.

B. Network Topology

In DFL, the network topology describes the organization of participants and determines their communications. Therefore, the importance lies in the impact on convergence, generalization, overhead, and robustness of the DFL approach. There are three network topologies used in DFL: fully connected networks, partially connected, and node clustering. Figure 6 shows these network topologies and their characteristics. In particular, the topologies are evaluated depending on the importance (high, medium, low) of robustness, flexibility, fault tolerance, communications cost, and security.

1) *Fully Connected Networks:* This network topology maintains direct links between all pairs of nodes. When a new node is added to the system, a link must be added to each node (see Figure 6a) [10]. Thus, the communication cost is high and grows with the number of nodes in the network. Moreover, adding new nodes increases the complexity of managing connections for each node, resulting in low flexibility. Despite the high communication cost and low flexibility, this topology is highly reliable and robust, as the network can still function when a few nodes or links fail [51].

2) *Partially Connected Networks:* In some network topologies, participating nodes may only have direct links to some nodes in the network, regardless of the shape of the topology. Therefore, the basic transmission cost for each node is lower than in fully connected networks since the model parameters are not transmitted to all nodes. As a result, communication cost may be higher than in fully connected topologies since the model parameters transmitted may go through several intermediate trainer participants until they reach the aggregating participant, resulting in a longer delay [55]. This work divides partially connected networks into star-structured, ring-structured, and random.

In a star-structured network, one of the nodes acts as a proxy participant and enables the federation communication with all other nodes (see the first subfigure of Figure 6b). When a new participant is added, only one link is needed to connect them to the central node. Consequently, the communication cost grows

linearly as the number of nodes increases, and the resources of the node limit network scalability. However, since all communications between non-central nodes must pass through the central node, it becomes a potential bottleneck, reducing flexibility. Additionally, the network has low fault tolerance, as the failure of the central node disrupts the communication between all nodes. Some DFL solutions use temporary connections in star topologies, but establishing and terminating these connections adds significant overhead.

In a ring-structured network, nodes are connected circularly (see the second subfigure of Figure 6b). The communication cost grows linearly as the number of nodes increases, as each node only maintains two fixed links, regardless of network size. This contributes to medium flexibility. However, as the number of nodes grows, transmission delays for model parameters also increase. Ring networks can be unidirectional, where nodes transmit model parameters to only one neighbor, or bidirectional, where nodes send parameters to both neighbors, providing higher reliability and fault tolerance than unidirectional rings.

Sometimes a defined topology is not supported by the federation, and nodes generate connections following heuristics based on the proximity or computational capacity of nearby neighbors. Due to intermittent connections in DFL, the network maintains a dynamic node structure, creating random network topologies over time [53]. In this sense, some topologies use expander graphs and deterministic local optimization algorithms to reduce communication overhead in these networks. The third subfigure of Figure 6b represents a network topology with six random nodes. Specifically, it describes a network based on the Erdős-Rényi model, where links between nodes are established randomly [54]. This type of network offers high flexibility and moderate fault tolerance but can result in higher communication costs and variable security levels.

3) *Node Clustering:* Recent publications have identified another DFL network topology for creating hierarchical clusters with models adapted to the distribution of nodes. In this regard, the literature addresses two approaches: similarity-based clusters [120] and proxy-based clusters [44]. In the former, clusters are determined by the similarity of the local model parameters of the nodes. As a result, each cluster is

more individualized for the nodes that compose it, obtaining a homogeneous performance between nearby nodes. In contrast, proxy-based clusters aim to use nodes that interconnect different topologies, forwarding model parameters. In this regard, proxy nodes have to be evaluated in terms of performance since they transmit data from one cluster to another, generating an important bottleneck in the overall architecture [58]. The disadvantage of this clustering method implies that all nodes are initially linked to a coordinating node, which is not feasible in all application scenarios. Furthermore, since nodes within a cluster share similar data distributions, individually trained cluster models may be less generic and robust than models exposed to global data distributions [57]. Figure 6c shows the usage of proxy nodes to interconnect topologies in node clustering.

C. Communication Mechanisms

This fundamental defines the procedures that enable DFL communications between network nodes. First, the communications scheme in DFL, based on synchronous, asynchronous, and semi-synchronous communications, is detailed. Then, this subsection presents two different decentralized communications mechanisms: peer-to-peer (P2P) [62] and gossip [121].

1) *Communications Scheme*: The communications schemes determine the behavior of the nodes when transmitting or receiving model parameters from neighbors and then aggregating them. Depending on the type of scenario and the objective pursued, DFL can be supported by synchronous, asynchronous, or semi-synchronous communications. The first alternative requires each node to perform local training, consisting of multiple steps, usually expressed in epochs. An epoch is defined as a complete iteration through the entire local dataset. After completing local training, all nodes acting as trainers send the parameters of their local models to all their neighboring nodes and receive the new model they have to integrate. This training technique, known as synchronization points, is repeated for several federation rounds. In this sense, the nodes will have to wait for the end of the round to transmit their model parameters again. Therefore, a drawback of this scheme is its slow convergence due to waiting for slow or idle participants. To address this problem, over-selection techniques could increase participant selection but also lead to resource waste and selection bias [11].

In asynchronous communications, no synchronization point exists, and nodes can transmit and receive model parameters independently of the other federation participants. Because there is no idle time for participating nodes, asynchronous protocols have a faster convergence speed [61]. However, they cause higher communication costs and lower generalization due to staleness [60]. It is also necessary to detect stragglers (i.e., slow devices) in the federation, delaying data propagation through the network topology. In the same way, aggregators need aggregation policies adapted to the number of model parameters received, as they vary during federation rounds. Nevertheless, asynchronous communications are well suited for cross-device DFL scenarios, in which nodes have varying computational capability and intermittent availability [59].

Finally, semi-synchronous communications provide a balance between resource usage and communication costs. In this schema, each node trains locally until a predefined synchronization point is reached. Consequently, nodes with varying processing capacities and data volumes perform different numbers of epochs. The batch serves as the basic computation unit, allowing for finer control over when a node contributes to the federation. Some studies, such as [60], introduce thresholds for each device that compare the changes in local model parameters with available local resources to determine the benefits of aggregation at each round.

2) *Peer-to-Peer Communications*: Traditionally, the deployment of overlay networks has been used mainly for P2P message exchange, online social networks, and routing infrastructures. DFL leverages P2P message exchange networks for local model parameter exchanges at each node. However, unlike traditional P2P networks, DFL deals with a dynamic and heterogeneous topology, where participants often change their location or role in the federation [62]. In this regard, selecting d random neighbors to exchange information is not always feasible. The main drawback of DFL is that nodes cannot randomly choose d neighbors among existing nodes since there is no central coordinator to determine these associations [67]. However, each node can determine neighbors if global information is given when the topology is created, and neighbor discovery mechanisms are implemented during federation [64]. Other alternatives are to employ distributed Delaunay triangulation networks for wireless sensor networks or regular random topologies for data center and memory interconnection networks [122].

3) *Gossip Communications*: Gossip communications enable asynchronous transmissions of model parameters between network nodes. This type of communication relies on the P2P sampling service with the objective that nodes in the federation communicate and exchange data [71]. The comparison of the DFL approach using gossip communications versus the CFL approach has been extensively studied, with superior performance obtained in the former when a better understanding of model parameters is employed [69]. However, this mechanism maintains a restrictive functionality where communication occurs between close neighboring nodes. To address this limitation, additional processes are presented to determine the location of a node using random walks [123] and node selection techniques, relying on the heuristics of the distance between nodes or underlying device features [72]. For instance, local models can be updated with partially received parameters and be optimized by mixing weights according to the matrix of communication link reliability [89]. Finally, some application scenarios of gossip communications in DFL are providing accurate recommendations (e.g., location logs, movie ratings) [72] or in Distributed Online Social Networks (DOSN) where users can directly communicate only with their immediate neighbors in a social graph [73].

D. Security and Privacy

Decentralized scenarios may be more susceptible to attacks than CFL scenarios as the number of participants grows within

the federation. Nevertheless, the level of vulnerability largely depends on the network topology. For example, in a fully connected network, where each participant has numerous connections, the impact of an attacker might be less severe than in a centralized CFL scenario since the attacker can only compromise a limited portion of the network. Nonetheless, decentralized scenarios still face increased risk due to the numerous intermittent and often weak connections between the participants.

In this sense, DFL can be subject to various attacks, including malware deployment and adversarial attacks, which can affect federation models and decentralized data, similar to the CFL approach [76]. While the former exploits intrinsic DFL fundamentals in the architecture or communications to affect the federation adversely, the latter aims to infer federated models and local node data, manipulate preconditions or destroy the model. Notable among these attacks are model inversion or membership inference, which can potentially infer the raw data by accessing the model [49]. Despite these potential attacks, countermeasures have been adopted to address them, preserve data and model privacy, and develop recovery plans to minimize harm and disruption. In particular, this paper presents two main approaches adopted in current DFL scenarios for data protection: cryptographic methods and differential privacy.

1) *Malware Attacks*: Malware, also known as “malicious software”, is a critical issue in the DFL approach due to the fundamentals based on decentralized architectures without a server coordinating the scenario. Malicious entities with high propagation through the network can affect behavior and interfere with its proper functioning. This issue differs in impact compared to more traditional approaches such as CFL, mainly because DFL does not have a centralized entity to coordinate the cooperation of participants in the federation. Thus, malware can be created and deployed directly among new network participants. Due to the number of connections to neighboring nodes, fully connected networks, star-structured networks, and random networks are the most affected in terms of security. While the first ones maintain communication links between each pair of nodes, the others establish links with either a central node or random nodes that could be malicious or dishonest.

Regarding communications between nodes, malware also has a significant impact due to the high number of exchanges in DFL. The exchanged messages are obtained by neighboring participants, extracted, and processed for aggregation or forwarding in the case of proxy participants. However, the information obtained may have been modified during the exchange, adding malicious data using rootkits or including the participant in a botnet [75], [76]. In this way, the node would be infected and could spread easily and quickly among the federation participants, especially when there are many links between neighbors.

2) *Adversarial Attacks*: DFL must address adversarial attacks caused by malicious nodes participating in the federation. These attacks, inherited from traditional CFL operations, include poisoning attacks on HFL and VFL approaches [49]. Furthermore, DFL is prone to poisoning attacks for the following reasons: (i) DFL scenarios involve numerous participants,

increasing the likelihood of faulty behavior from one or more nodes; (ii) local training data and training processes of participants are hidden from other nodes in the network, making it impossible to verify the authenticity of updates sent by participants; and (iii) multiple participants may generate vastly different local updates. Consequently, secure aggregation protocols might be negatively affected by aggregation performed by malicious nodes, rendering it infeasible to audit local updates [56]. In this sense, the paper at hand analyzes such attacks on DFL solutions to assess their impact on the network.

Adversarial attacks can be classified based on the timing of the attack (training or evaluation time) and frequency (one-shot or multiple) [124]. Among training-time attacks, Byzantine attacks are prevalent. In these attacks, malicious nodes disrupt model training by transmitting poisoned, corrupted, or fake model updates to other nodes in the network. Since malicious nodes in DFL keep their data and aggregations private, they can poison their model and distribute their parameters without consequences. These attacks can be executed in a one-shot manner each round or multiple times in asynchronous communication scenarios. Similarly, Sybil attacks create multiple fake pseudonym nodes that flood DFL networks. It is difficult to mitigate these threats since no central authority can verify node identities and distinguish between pseudonym and non-pseudonym nodes. Using pseudonyms, an attacker can launch DDoS attacks and disrupt the DFL network. The attacker can command multiple pseudonyms to flood a targeted node with pairing requests, rendering it unable to respond to and communicate with legitimate nodes [80]. Additionally, an adversary can corrupt the reputation system of decentralized applications. Nodes could obtain a false high reputation, being forced to participate in the network and being queried by other nodes, thus degrading their performance.

Other attacks attempt to infer DFL communications with a high attack frequency. One of these threats is Eclipse attacks, which enable attackers to gain partial or full control of a participant by manipulating its connections with neighboring nodes [78]. In such attacks, the attacker nodes control many connections, which allows them to “eclipse” the victim, rendering messages that should reach it useless. The attackers could gain complete control of all traffic to the victim if all node connections belong to them [79]. Free Rider attacks are also a type of attack that takes advantage of DFL without contributing useful data to it during the training process [125]. Finally, in evasion attacks, malicious nodes mimic participation in DFL by generating misleading updates, crafting inputs that manipulate the model into producing faulty collaborative models.

3) *Security Countermeasures*: The use of countermeasures is essential in the DFL approach to prevent malicious participation and preserve data privacy. DFL establishes a federation by sharing model parameters, such as gradients, weights, or meta-level information among participants. However, exchanging gradients or meta-level information during distributed training can reveal information about the training data and model updates, making exchanging model weights preferable for privacy preservation [81].

Additional countermeasures comprise several techniques inherited from CFL architectures, as reported in [12]. Most of them are employed during decentralized federation to obfuscate updates at the cost of reducing the accuracy of each local model, relying on traditional Differential Privacy (DP) relaxations to inject less noise. However, this technique may degrade the performance of constrained participants, leading to suboptimal results in the federation. In DFL, each participant needs to perform a significant computation to add or suppress noise per set of model parameters transmitted on each communication link, making it more challenging to adapt than CFL [82]. Other techniques, such as data augmentation and obfuscation, can prevent sample reconstruction in local training. However, model parameters shared among neighbors would be equally exposed to malicious nodes. A promising solution is to combine DP mechanisms with secure aggregation and additive Homomorphic Encryption (HE), depending on the application scenario and required security level, and deploy them on a larger number of federation participants. In other cases, it is preferable to rely on anomaly detection mechanisms [126] or model misbehavior detection [90] to detect malicious actors that may deteriorate DFL performance. While the former focuses on the data supplied to the model, the latter analyzes the internal performance of the model depending on the task. In these cases, detection could involve the deployment of mitigation to reduce the severity of a cyberattack. Recent research has further augmented defensive capabilities by integrating Moving Target Defense (MTD) techniques. Specifically, random neighbor selection and dynamic IP/port switching offer enhanced resilience against communication-based threats [83]. Moreover, anomaly detection has significantly improved by incorporating time series analysis techniques [84].

Finally, cryptographic methods, such as Secure Multiparty Computation (SMC), are widely used in DFL to preserve the privacy of exchanges between neighboring nodes [45]. Federation participants must encrypt their messages before sending them, operate on them, and decrypt the encrypted output to obtain the final result. Despite the security provided by SMC, nodes can experience a significant computational overhead, similar to previous DP techniques. To address this issue, new approaches have emerged to provide adequate security performance while maintaining efficient resource usage. One such approach is based on deploying a Trusted Execution Environment (TEE) [77], such as Intel SGX processors, which can protect the code and data loaded inside it. Each node in the network can use this environment to increase its trustworthiness while allowing model parameters to be securely added during federation. In addition to cryptographic mechanisms and secure environments, DFL also requires mechanisms that guarantee the decentralization of the architecture while ensuring inter-participant reliability. DLT technologies, such as Blockchain, provide an ideal solution for this requirement by ensuring the immutability of exchanged models [68]. It is accomplished through a P2P consensus scheme governing the network, where the model parameters of each participant can only be updated through consensus. In this sense, a novel solution uses a lightweight Blockchain with

a unique consensus protocol and storage strategy, significantly reducing overhead while guaranteeing privacy. This approach optimizes performance by mitigating slower device impacts and maximizing storage utilization [127]. At the same time, a transparent and immutable reward system ensures the trust of the participants by detecting malicious activities.

E. Key Performance Indicators

KPIs are necessary to measure the efficiency and suitability of the activities and components of DFL architectures. It is essential to know these metrics to detect and resolve federation, communications, or security issues more quickly, reducing the frequency and duration of incidents while increasing efficiency [86]. In this sense, KPIs are utilized for the proper evaluation of the main components of the DFL approach: (i) the federation nodes, detailing the evaluation in terms of mobility and resource capabilities; (ii) federation communications, defining the flexibility and overhead of the deployed architecture; and (iii) the federation models, reviewing their performance and trustworthiness in solving distributed ML tasks (see Figure 7).

1) *Federation Nodes KPIs*: DFL must have KPIs to evaluate the participating nodes of the federation. Thus, the heterogeneity and dynamism of the nodes in DFL scenarios are relevant factors adding complexity to the network. In this sense, *resource capabilities* and *node mobility* are identified as promising KPIs to determine node performance. The *resource capabilities* KPI comprises computational and network capacity.

- *Computing capacity*: It provides internal participant resource indicators for local training computation and model parameter aggregation during federation. Some metrics related to this category are CPU/GPU, disk, or RAM usage. One of the aspects of computational capacity is the autonomy of the node to address the tasks. In this sense, time, cycles, and temperature are considered [87].
- *Network capacity*: It determines the node ability to manage communications with one or more neighbors. In this regard, the following metrics are important to analyze: the number of bytes transmitted and received, the number of packets transmitted/received, the bandwidth, or the number of listening/established sockets [7].

In addition to *resource capabilities*, DFL presents the *node mobility* KPI to evaluate the dynamism of the network. Therefore, nodes can be easily affected in communications with other nodes, generating intermittent inputs and outputs in the network. The *node mobility* KPI is determined by node traceability in both the network and local.

- *Network traceability*: It monitors the state of each node and traces its mobility in the network. For this purpose, this metric computes the edges of nearby participating nodes to determine their link with other participants in the federation, which denote the number of near neighbors. In this way, the network can extract information about the position, direction, and absolute velocity of each node [13].

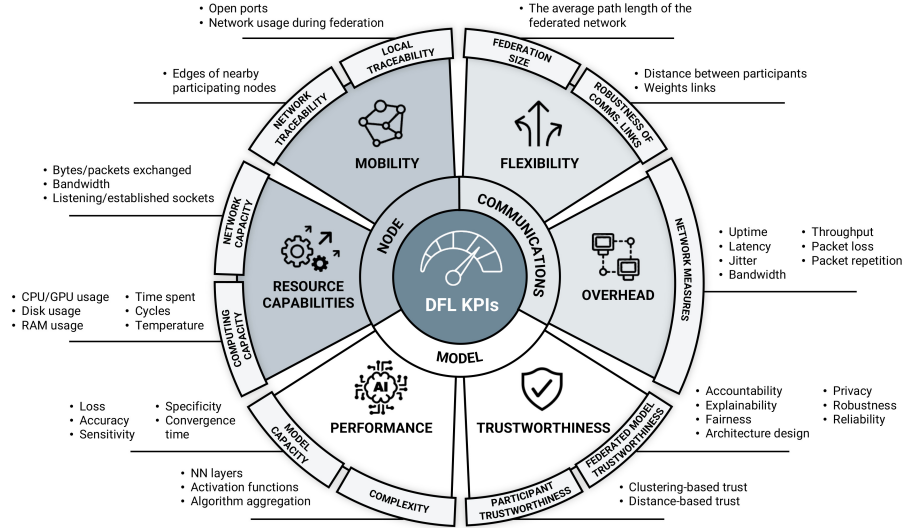


Fig. 7. Key Performance Indicators for DFL.

- **Local traceability**: It allows determining the DFL network to contribute by monitoring local interfaces and their characteristics. Thus, it is possible to determine the mobility intent. Metrics such as the number of open ports or network usage during federation are identified [128]. Network traceability metrics usually complement them in the evaluation of node performance.

2) **Federation Communications KPIs**: Model parameter exchanges between nodes in a DFL approach might experience instability due to asynchronous communications, leading to nodes processing and exchanging data at varying rates. Consequently, this could result in delays and inconsistencies in the learning process, as some nodes may handle outdated information. Furthermore, the aggregation process is complicated by heterogeneous aggregator devices with diverse computational capabilities and communication protocols. Such heterogeneity can cause inconsistencies across different network segments, ultimately having a negative impact on the accuracy of the models for a group of participants. Moreover, asymmetric topologies may cause uneven data distributions or processing capabilities among nodes, posing challenges in attaining a uniformly distributed learning process. In this sense, *communications flexibility* and *network overhead* are essential for optimal balance. Regarding the *communications flexibility*, the DFL approach is evaluated according to the following elements.

- **Federation size**: It provides an evaluation of the predisposition of the architecture to exchange model parameters between nodes. This metric accurately calculates the average path length of the network, representing a stable number strongly affected by the movement of the nodes and, thus, the flexibility of the network. Reduced network size improves parameter transmission performance by reducing interference [10].

- **Robustness of communication links**: It determines the strength of the links with the other neighbors participating in the federation. Some metrics that allow its evaluation are the distance between nodes, reactance, and reliability. Link weights are often used to support evaluations, with high weights being maintained for links with a high percentage of correctly received transmissions [89].

In the context of DFL, several scenarios require decentralized technologies, which can face significant performance issues. One of the main challenges arises from the communication-intensive nature of DFL applications, which require frequent model parameter transmissions involving large amounts of data. As the federation progresses or the model becomes more complex, this communication becomes even more intensive, leading to high *network overhead*, where a high overhead can lead to applications failing to meet response time, quality, or resource utilization requirements. *Network overhead* is evaluated from the following metrics [7].

- **Uptime**, often known as availability, reflects the time of use of the federated network since it became available to participants.
- **Latency** measures the time a data packet travels between network nodes.
- **Jitter** indicates the variability of a latency time and measures the consistency of a network transfer rate. Very low jitter is preferred when a real-time reaction is required (e.g., warning of enemy UAVs).
- **Bandwidth and throughput**. While the first indicates the quantity of data a network path is anticipated to support or successfully convey from one point to another at a specific time, the second describes the amount of data transported between nodes inside a network path.
- **Packet loss** can denote congestion, low bandwidth, and interference. It refers to packets transferred from one

TABLE III
COMPARISON OF THE FLEXIBILITY OF FEDERATED COMMUNICATIONS
BETWEEN DFL, SDFL, AND CFL

Federated Architecture	Communication Times/Round	Node Overhead	Total Transferred Data Volume per Round (MB)
DFL	1	$M \cdot (N - 1)$	$M \cdot (N - 1)^2$
SDFL	$\left\lceil \frac{N-1}{2} \right\rceil$	$2 \cdot M$	$2 \cdot M \cdot N \left\lceil \frac{N-1}{2} \right\rceil$ *
CFL	1	$M \cdot N$	$M \cdot N$

* It considers the additional communication required for leadership transmission, which is not present in the DFL and CFL architectures.

device to another but fail to arrive at their destination. Packet loss is the ratio of packets received at the destination to packets sent from the source.

- *Packet repetition.* When a data packet is successfully sent but does not reach its destination, it must be retransmitted. Retransmissions occur in all networks; however, they are more common in wireless networks due to low signal strength, concealed nodes, and interference.

Both *communications flexibility* and *network overhead* are mainly limited to the model parameters exchanges during federation for each K step. Assuming M is the parameter size of the model, there would be $N - 1$ times communications in a round, being the average overhead $M \cdot (N - 1)$. Thus, K could reduce the communication frequency, decreasing the performance of DFL. A summary of the communication complexity comparison with CFL architectures is shown in Table III, where $\lceil \cdot \rceil$ denotes the standard rounding function. The table shows that the total volume of data transferred per DFL round is comparable across all three architectures. However, the communication overhead per node is different among decentralized architectures. Specifically, SDFL has the potential to achieve better performance in node communication overhead, which could benefit system bandwidth utilization and increase robustness. Overall, it is important to consider the trade-off between communication overhead and other features when designing decentralized architectures with FL.

3) *Federation Models KPIs:* The *performance* of the federated model in solving the task for which it has been created is critical. In other words, the performance indicates the success of DFL in generating accurate predictions or solving the intended problem. The following elements are used for measurement [90].

- *Model capacity:* It is based on comparing the model predictions with known values of the dependent variable in a dataset. Some relevant metrics are model loss, accuracy, sensitivity, specificity, and the time associated with the convergence of the model.
- *Complexity:* It is influenced by both the functions of the model trying to learn and the nature of the training data. Some metrics are the number of layers in the NN, activation functions, and algorithm aggregation implemented by the federated model.

In addition to the above elements, federation models must be *trustworthy* to ensure the technology is used to contribute productively to the task. If they are not trustworthy, nodes will not

collaborate with these systems. The problem increases when model parameters are deliberately distributed among nodes in a topology. Therefore, it is necessary to determine the confidence of the participants and the federated models to ensure trustworthiness, where robust privacy and security measures significantly contribute to this [92], [93].

- *Participant trustworthiness:* It defines the confidence level of neighboring participants based on the model parameters exchanged. There are two main metrics, clustering-based trust, which determines the federation nodes trustworthiness relying on clusters of participating nodes; and distance-based trust, which is determined based on the distance between participants and the transmission time of the model parameters.
- *Federated model trustworthiness:* It provides confidence in the local models of each participant in the federation. Normally it is calculated in the aggregator nodes before supplying the federated model to the rest of the participants. The metrics are divided into several pillars: accountability, analyzing the quality of the model life cycle (client registry, anomaly monitoring); explainability, providing clarification of the decision rationale (algorithm class, model size, DFL configuration); fairness, ensuring impartial and just decisions without discrimination of protected groups (participation rate, discrimination index); architecture design, creating a baseline behavior for the management of participants in the federation (data augmentation, regularized local loss); privacy, protecting federated model data (perturbation, anonymization, entropy); robustness, checking the resilience against adversarial inputs (confidence score, loss sensitivity); and reliability, predicting when an asset will fail or deteriorate (client dropout rate, data provenance, data quality).
- *Security and privacy:* The preservation of security and privacy is paramount in DFL, even while allowing effective model training [95]. DP is a widely used method that includes a privacy budget parameter, denoted as epsilon (ϵ), serving as a metric to quantify the maximum amount of privacy loss allowed. This budget diminishes with each data-sharing instance, with smaller budgets enhancing privacy but potentially impeding data utility. Another critical parameter in DP is delta (δ), which represents the probability of information inadvertently being leaked. Other privacy-preserving methods, such as HE and anonymization, also have vital performance indicators. In HE, the degree of noise introduced can be a significant parameter. Anonymization techniques, such as k -anonymity and l -diversity, use k and l values as performance metrics, ensuring privacy in DFL by making it difficult to re-identify models within a federation [96].

F. DFL Optimizations

Despite the advantages of DFL, a series of optimization mechanisms are necessary to guarantee the full efficiency of the approach. In this sense, DFL includes three main elements to optimize: federation nodes, federation communications, and federation models (see Figure 8).

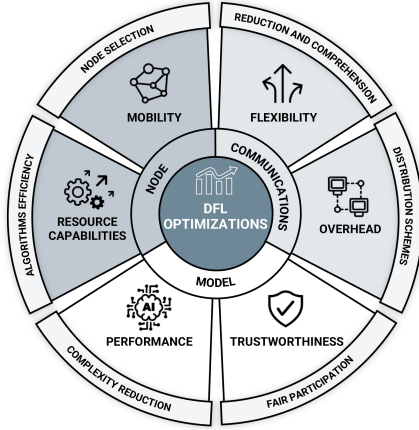


Fig. 8. DFL optimizations.

1) *Optimization of Federation Nodes*: An important optimization for improving node *resources capabilities* and *mobility* in the federation is the selection of nodes to transmit the model parameters. Node selection is divided into three different strategies.

- *Sequential selection of nodes*: Participants are selected at the beginning of the federation process, which iterates over time [93].
- *Random selection of nodes*. It excludes resource capabilities or network constraints from the participants. In this sense, a subset of nodes is obtained in a fixed time of the federated process to send the model parameters.
- *Scheme schedule selection of nodes*: This selection strategy requires a node to act as an orchestrator of the communication links and the characteristics of each neighboring node. Furthermore, the controller can select participants based on a preassigned probability calculated according to participation in the federation [4].

Additionally, it is relevant to optimize traditional aggregation algorithms such as Federated Averaging (FedAvg) to make it more suitable for the node resource capabilities [4]. FedAvg is a widely accepted heuristic algorithm used as a baseline for star-structured topologies due to its simplicity and empirical effectiveness. In this case, the aggregator node obtains the model parameters of the neighboring nodes, aggregating the average of the updates and computing the resulting global model. It is also worth noting that while in CFL, the FedAvg algorithm has been widely accepted as a baseline, in DFL, no algorithm has risen among the others. Customizations of FedAvg and new techniques focused on decentralized scenarios have emerged in the literature intending to adapt to DFL solutions.

- *Decentralized Stochastic Gradient Descent (DSGD)*: It provides optimization for a set of objective functions distributed over a network. Each node maintains estimates of the optimal optimization step on information concerning its cost function and exchanges these estimates directly with the other nodes in the network [129].

- *FedPGA*: It defines a partial gradient exchange mechanism that leverages node-to-node bandwidth to speed up communication time. At the same time, it reduces the convergence rate by adapting the step size of the stable gradient descent direction [97].
- *Dynamic Average Consensus-based FL (DACFL)*: It transforms the aggregation of the DFL model into a local training procedure as a discrete time series, whose convergence guarantees convex objectives [130]. In the algorithm, nodes compute and update a local solution to a subproblem and share the solution with neighboring nodes. Moreover, the authors demonstrate that the algorithm adapts to dynamic networks and heterogeneous scenarios by tuning a node-specific local parameter based on the node resources [5].
- *Split Learning (SL)*: This method, called SplitNN [99], enables collaborative training of NNs without sharing raw private data, but emphasizes the suitability for DFL architectures. It employs split models instead of full model replication in a DFL architecture. The training participants hold replications of the shallower layers up to a certain layer (i.e., the cut layer), while a specific node holds the deeper layers.
- *Decentralized FedAvg with Momentum (DFedAvgM)*: It uses SGD with momentum to train the local models of the federation participants, communicating only with their neighbors in an undirected graph. It provides a better convergence rate and communication efficiency than DSGD [131].
- *DeceFL*: It focuses on neighbor communications to improve efficiency for both convex and non-convex loss functions. It applies to a wide range of real-world medical and industrial applications [132].

2) *Optimization of Federation Communications*: Optimizing network flexibility and overhead is crucial for addressing performance issues caused by continuous data exchanges between participants. Concerning *network flexibility*, it is necessary to have techniques that reduce the number of exchanges in the network without losing functionality during federation [133]. Several studies have proposed reducing the number of bits transferred for each worker update through data compression. However, these studies do not consider the impact of data loss from compression results, which could impair the learning accuracy and affect convergence [103].

In the literature, different distributed optimization schemes aim to maintain acceptable convergence rates in terms of recurring iterations and device computation time. Some examples are Decentralized Gradient Descent (DGD) [104], decentralized Alternating Direction Method of Multipliers (ADMM) [105], EXTRA [106], and ADMM based on Jacobi-Proximal [107]. A recent contribution includes a method that applies the Lloyd-Max algorithm to DFL to minimize quantization distortion, with the resulting LM-DFL algorithm able to adjust quantization levels adaptively [102]. Other communication optimization mechanisms use an incremental learning method to reduce costs by activating and linking agents while keeping other nodes and links inactive.

These include Random Walk ADMM (WADMM) [134], Parallel Random Walk ADMM (PW-ADMM) [123] and Walk Proximal Gradient (WPG) [135] which are commonly used in increment-based approaches. Recent research has introduced mechanisms that aim to optimize the balance between improving the quality of the model and saving communication resources, which can result in a more sustainable federation policy [100]. These mechanisms typically offer an independent selection of participants and fragments of the NN to be transmitted, providing a promising alternative to traditional optimizations. One such technique is FL-EOCD, leveraging D2D communications and overlapped clustering to enable decentralized aggregation, thereby reducing overall energy consumption and latency while maintaining convergence rates [58]. Further approaches, such as [136], implement latency optimization strategies by incorporating Blockchain. The latency optimization is achieved by strategically managing data offloading decisions, power transmission, bandwidth allocation, and computational resources, resulting in a highly responsive system.

Regarding optimizing *network overhead*, the DFL approach presents limitations in communication bandwidth between nodes [71]. To address this issue, researchers have explored different distribution schemes, such as quantization and sparsification. For example, Quantized Sparse SGD (QSGD) [109] and quantized ADMM [137] were proposed to reduce network overhead when exchanging model parameters. Furthermore, the Qsparse-local-SGD algorithm [110] includes a combination of aggressive sparsification with quantization and local computation along with error compensation. However, these methods may decrease accuracy to achieve lower communication costs [74]. With adequate communication infrastructure, decentralized learning can achieve accuracies like its centralized counterpart. In this sense, the use of advanced technologies such as 5G [138] or 6G [139] can help reduce communication limitations, improving network bandwidth and enable faster and more efficient data exchange. Other studies, such as [140], have shown that decentralized SGD can outperform centralized approaches in low-bandwidth networks. However, these studies do not consider typical federated settings, such as non-IID data among participants, limiting the applicability of their results.

3) *Optimization of Federation Models*: It is necessary to optimize federated models to avoid high processing demand on the most constrained participants. In this sense, optimizations are responsible for increasing the efficiency of the models without sacrificing performance and trustworthiness.

In DFL, nodes often aim to train a state-of-the-art ML model on a specified task. To increase *model performance* and *trustworthiness*, researchers have developed new models and improved existing solutions. The most popular ML model is a NN, which achieves state-of-the-art results in many tasks, such as image classification. Many DFL studies are based on Stochastic Gradient Descent (SGD), which can be used to train NNs [110]. Another widely used algorithm is a decision tree, which is more efficient to train and easier to interpret than NNs. In particular, a tree-based DFL is designed for federated training of single or multiple decision

trees, such as Gradient Boosting Decision Trees (GBDTs) and Random Forests (RFs). GBDTs have been especially popular recently due to their superior performance in many classification and regression tasks [26]. Besides NNs and trees, linear models such as Linear Regression (LR), Logistic Regression, or Support Vector Machine (SVM) are classic and easy-to-use models. To enhance the consistency of models generated by each federation participant, algorithms such as DFedSAM utilize gradient perturbation to produce local flat models with uniformly low loss values [111]. In non-IID scenarios where adaptive training models can be challenging on resource-constrained devices, these models can be particularly useful [108]. Furthermore, techniques such as SwarmSGD jointly leverage non-blocking communication, quantization, and local steps, allowing for better compensation by training on more samples but with a lower quantization level of model parameters. This technique is effective in heterogeneous node data distributions and random topologies [88].

In addition to the efficient models mentioned above, new techniques have emerged in recent years to improve the efficiency and performance of DFL. These techniques include meta-learning, multitasking, and federated distillation. Meta-learning aims to improve the efficiency and performance of the model generated at each node by creating less complex models capable of performing tasks in parallel. It uses information from the activities performed to better adapt to new tasks [112]. As a result, meta-learners generalize better when trained with a more extensive set of practical tasks and more data for each task. In this sense, the learner can acquire knowledge and processing capacity distributed among agents, adapting its model to the best-performing ones in the network. Meta-learning often involves multitasking when ML models are trained to perform multiple tasks simultaneously. These models optimize metrics for each task and generate a corresponding output [91]. Finally, federated distillation, first introduced in 2015 as knowledge distillation, aims to optimize federated models by performing model compression. Specifically, it is a procedure to train a model using a second NN as a reference instead of learning directly from the ground truth of the data [141]. In this way, it allows several advantages over more complex models: (i) more information can be extracted from a single sample, (ii) training can be performed with fewer examples, and (iii) labeled data are not needed. All this offers less computation time and energy consumption in the federated node [113]. A decentralized version of this approach has been proposed in [114], where instead of having a pre-trained model to imitate it, each model considers the aggregated knowledge of the remaining participants' models.

Finally, the most relevant optimization of federated models for increasing *trustworthiness* is using DLT techniques, providing reliability during model parameter updates in the federation. They also facilitate the fair participation of nodes during the federation. In such trust-based relationships, there are two entities: the trustor and trustee node. For each observation from the trustee, the trustor determines whether it is an outlier, while the trustee node validates the contributions provided by the trustor. To maintain the reputation of the participating devices and ensure that all participants make active

and honest contributions to the model, the trustor node i can generate a smart contract with the parameters of the local model, which is validated by a set of trustee nodes j [142]. DLT techniques also provide an added explanation and robustness to federated models. The immutability of the initialization and update of federated models is guaranteed by DLT, maintaining a chronological log in the Blockchain. Moreover, the fair participation of nodes in the federation is further ensured by DLT.

IV. OPEN-SOURCE FRAMEWORKS FOR DFL

Building upon the storyline presented in Figure 2, this section responds to “Q2. What DFL frameworks exist, and what fundamentals do they provide?” In this context, frameworks serve as the bridge between the “Theoretical Overview” with its fundamentals and “Practical Application” with its application scenarios. This section introduces the most relevant existing open-source frameworks for DFL. First, it discusses the most mature and highly customized frameworks supported by large companies. Then, it focuses on those more incipient with limited functionality but notable research advances. Table IV compares the main aspects of the frameworks analyzed. The double horizontal line separates the mature frameworks from the incipient ones.

A. Mature Frameworks

TensorFlow Federated (TFF) [143], developed by Google, provides the building blocks for FL based on TensorFlow. It includes three key components: models, federated computation builders, and datasets. TFF can be deployed on multiple machines, creating a rotating aggregator. Currently, TFF does not consider any adversaries during FL training nor provides privacy mechanisms. Other frameworks, such as PySyft [144], provide interfaces for developers to implement their training algorithm. While TFF is based on TensorFlow, PySyft can work well with PyTorch and TensorFlow. This framework provides multiple optional privacy mechanisms, including secure multi-party computation and differential privacy. Moreover, it can be deployed on a single machine or multiple machines, where the communication between nodes is performed using the WebSocket API. For privacy preservation, SecureBoost [56] builds multi-part reinforcement trees with an encryption strategy between different nodes. This solution allows for different feature sets corresponding to a vertically partitioned dataset. One of the advantages of SecureBoost is that it provides remarkable accuracy across different topology architectures without revealing information about each private data provider.

Other frameworks, such as FederatedScope [145] and FedML [146], allow lower-level management of the deployed architecture. The first one employs an event-driven architecture to provide great flexibility in describing node behaviors. It allows learning objectives and backends to be added and coordinated in an FL life cycle with synchronous or asynchronous training strategies. In addition, it includes privacy protection, attack simulation, and automatic tuning. In addition, FedML offers a platform for FL benchmarking based

on PyTorch and utilizes the Message Passing Interface (MPI) protocol, gRPC, or MQTT for communication. This framework comprises two main modules: FedML-core, which implements the training engine and the distributed communication infrastructures, and FedML-API, which is built on top of FedML-core and provides training models, datasets, and FL algorithms. Finally, FedML supports three computing paradigms: standalone simulation, distributed computing, and on-device training. Along the same lines, LEAF [147] provides only benchmark functionality of the deployed architectures. In this sense, it uses distributed datasets and partitioning mechanisms that other frameworks can leverage to complement its functionality.

B. Incipient Solutions

BrainTorrent [148] is an FL framework without a central server explicitly aimed at medical applications. With no need for a central authority to oversee the training process, BrainTorrent provides highly dynamic P2P communications where each medical center communicates directly, also offering robust training of participants through asynchronous updates achieving performance similar to on-device model training. Although the framework focuses on image segmentation, it can be extended to other data and models. Similarly, Scatterbrained [43] facilitates the construction of both CFL and DFL systems. In the latter, it can create a customized communication system based on ZeroMQ, corresponding to a communications library based on UDP. Nodes can have a specific role: (i) leech, in which nodes listen for broadcasts but do not share data; (ii) offline, where nodes do not listen or broadcast data; (iii) peer, where participants listen and broadcast data; and (iv) seeding, where nodes broadcast but do not listen to incoming data. In addition, it allows the use of different ML frameworks such as TensorFlow, Scikit-learn, or PyTorch.

Other studies present DFL models based on the InterPlanetary File System (IPFS) [149]. In this sense, the authors deployed a framework called IPLS, whose nodes are interconnected and perform federated model training based on the operation of one or more network participants. Any participant can initiate the federated training process or join an existing one. This solution shows robustness in the face of dynamic nodes and intermittent connectivity, ensuring limited resource usage and convergence of the federated model. Similarly, TrustFed [142] uses IPFS and Blockchain smart contracts to maintain the reputation of participants in a centralized or decentralized network. In this way, compelling participants to make active and honest contributions to the federated model. In contrast, other frameworks such as FLoBC [150] deploy a Blockchain mechanism to ensure the decentralization of nodes. In this sense, the framework compares and contrasts the effects of the trainer-validator relationship, reward-penalty policy, and model synchronization schemes on overall system performance. The authors demonstrate that using this framework enables the deployment of DFL systems with performance similar to more centralized architectures. Similarly, BLADE-FL [151] uses Blockchain

TABLE IV
COMPARISON OF MATURE AND INCIPIENT DFL FRAMEWORKS

Reference	OS	Federation Participant Type	Architecture Aggregator Node	Aggregation Algorithms	Communication Protocol	Security and Privacy	Data Type	Scenario	Benchmarking
TFF [143]	Linux MacOS	Cross-silo	Centralized Decentralized	Median FedAvg FedProx	gRPC	✓	Time series Images	Simulation	✓
PySyft [144]	Windows Linux MacOS Mobile	Cross-silo Cross-device	Centralized Decentralized	FedAvg	Websockets	✓	Images	Simulation Real	✓
SecureBoost [56]	Linux MacOS	Cross-silo	Centralized Decentralized	FedAvg GBDT	gRPC	✓	Time series	Simulation	✗
FederatedScope [145]	Windows Linux MacOS Mobile	Cross-silo Cross-device	Centralized	FedAvg FedOpt	gRPC	✓	Time series Images	Simulation Real	✓
FedML [146]	Linux MacOS	Cross-silo	Centralized Decentralized	FedAvg FedOpt FedNova	gRPC MPI MQTT	✓	Time series Images	Simulation Real	✓
LEAF [147]	Linux MacOS	Cross-silo	Centralized Decentralized	-	-	-	Time series Images	Simulation	✓
BrainTorrent [148]	Windows Linux MacOS	Cross-device	Decentralized	FedAvg	N/S	✗	Time series	Simulation Real	✗
Scatterbrained [43]	Windows Linux MacOS	Cross-device	Centralized Decentralized	FedAvg	ZeroMQ	✗	Time series Images	Simulation	✗
IPLS [149]	Windows Linux MacOS	Cross-device	Decentralized	FedAvg	P2P	✓	Time series Images	Simulation Real	✓
TrustFed [142]	Windows Linux MacOS	Cross-device	Centralized Decentralized	FedAvg	P2P	✓	Time series Images	Simulation	✗
FLoBC [150]	Windows Linux MacOS	Cross-device	Decentralized	FedAvg	HTTP (REST API)	✗	Time series Images	Simulation Real	✓
BLADE-FL [151]	Windows Linux	Cross-device	Decentralized	Custom algorithm	P2P	✓	Images	Simulation	✓
DISCO [15]	Windows Linux MacOS Mobile	Cross-device	Centralized Decentralized	Custom FedAvg	peer.js	✓	Time series Images	Simulation	✗
CMFL [152]	Windows Linux MacOS	Cross-device	Decentralized	Median Trimmed Mean Krum Multi-Krum	P2P	✓	Time series Images	Simulation	✓
DeFL [40]	Windows Linux MacOS	Cross-device	Decentralized	Custom algorithm	N/S	✗	Time series Images	Simulation Real	✓
FL-SEC [12]	Windows Linux MacOS	Cross-device	Decentralized	FedAvg	N/S	✓	Time series	Simulation	✓
DisPFL [38]	Windows Linux MacOS	Cross-device	Decentralized	FedAvg Ditto FOMO Sub-FedAvg	P2P	✗	Time series Images	Simulation Real	✗
GossipFL [70]	Windows Linux MacOS	Cross-device	Decentralized	FedAvg S-FedAvg D-PSGD CHOCO-SGD	P2P	✗	Images	Simulation	✓
Fedstellar [153]	Windows Linux MacOS	Cross-silo Cross-device	Centralized Decentralized Semi-Decentralized	FedAvg Krum TrimmedMean Median	P2P HTTP (REST API)	✓	Time series Images	Simulation Real	✓

N/S (Not Specified) by the authors. The double horizontal line indicates the separation between mature and incipient frameworks

to mitigate vulnerabilities during federation. It offers robustness against malfunctions, distrustful connections, and external threats by having each client compete to generate a block before local training in the next federation round. The system also addresses training deficiency caused by lazy clients and optimizes resource allocation.

In addition to the above solutions, some frameworks explore new applicability technologies to generate DFL scenarios. DISCO [15] is a framework presented by the EPFL Machine Learning and Optimization Laboratory that creates an easy-to-use platform that allows non-specialists to participate in collaborative learning. This platform is based on Javascript, supporting arbitrary DL tasks and model architectures via TF.js and relying on P2P communications. Additionally, DISCO

provides mechanisms to ensure communication efficiency, privacy preservation, fault tolerance, and deployment customization. Moreover, frameworks seek to improve the robustness of their privacy techniques. In this case, [152] presented a serverless framework named Committee Mechanism-based Federated Learning (CMFL), orchestrated by a committee system deploying selection strategies of model parameters during federation. In contrast to the other frameworks, this one implements several aggregation algorithms such as median, trimmed mean, Krum, and Multi-Krum, evaluating them in terms of robustness and efficiency. Regarding the security and privacy of the scenarios, [12] and [15] provide frameworks capable of ensuring the integrity and confidentiality of the communications and data exchanged. Both alternatives

present custom aggregation algorithms to process image and time series data.

Finally, other frameworks in the literature allow the deployment of cross-silo-based solutions, creating robust topologies of nodes that exchange information using P2P networks. In this case, [40] removed the central server by aggregating the weights at each participating node, and the weights of only the current training round are maintained and synchronized among all nodes. This framework has been evaluated on two widely adopted public datasets, CIFAR-10 [154], and Sentiment140 [155], demonstrating convergence against more common threats with minimal accuracy loss, achieving up to 100x reduction in storage overhead and up to 12x reduction in network overhead compared to state-of-the-art DFL approaches. Similar to the previous solution, the authors of [38] allow further savings in communication and computational costs. In this sense, the authors proposed a decentralized sparse training technique, where each local model only maintains a fixed number of active parameters throughout the local training and P2P communication process. They employ aggregation algorithms such as Ditto, FOMO, and Sub-FedAvg to enhance efficiency. Furthermore, the authors demonstrated that the framework allows easy adaptation to heterogeneous local clients with different computational complexities. Similarly, GossipFL [70] presents a novel approach to address communication challenges in IoT scenarios characterized by intermittent connections between participants. The framework utilizes sparsification techniques and a unique gossip matrix generation method, employing algorithms such as S-FedAvg, D-PSGD, and CHOCO-SGD to optimize communication traffic while preserving convergence. The efficacy is demonstrated across various datasets, such as MNIST and CIFAR10, achieving model accuracies of up to 98%. Building on these advancements, Fedstellar [153] offers a versatile platform for DFL training that addresses common limitations in existing frameworks. It efficiently manages heterogeneous topologies and adapts the federation to physical or virtual deployments. In a real-world cyberattack detection employing single-board devices and a fully connected network, Fedstellar obtained a 91% F1 score. Furthermore, in a virtualized scenario using the MNIST dataset, the platform delivered a compelling F1 score of 98% with DFL and 97.3% with SDFL while reducing model convergence training time by 32% relative to CFL.

V. APPLICATION SCENARIOS

Following the storyline presented in Figure 2, this section responds to “Q3. Which are the main characteristics of the most relevant scenarios of DFL?” In particular, this section provides a review of existing works classified according to the application scenario: (i) healthcare, (ii) Industry 4.0, (iii) mobile services, (iv) military, and (v) vehicles. These scenarios combine most of the literature work, thus giving a great overview of current DFL applicability. For each scenario, the following aspects are highlighted: the goal, the participants during federation, the ML algorithm used, and the aggregation method employed.

Beyond the previously discussed scenarios, DFL is also relevant in emerging areas such as smart cities and metaverse, where it enhances urban planning and service provision, all while facilitating privacy-preserving data sharing between real and virtual entities [156]. In the realm of swarm robotics, DFL enables asynchronous knowledge sharing and task coordination [157]. In the energy and utilities sector, DFL supports predictive maintenance and demand forecasting, contributing to operational efficiency across companies [158]. Finally, the telecommunications sector benefits from DFL in optimizing network performance and anomaly detection, allowing network providers to share insights without compromising proprietary information [159]. However, the challenges of these domains need to be further explored to determine the scope of DFL.

A. Healthcare

Cooperation between research institutes, hospitals, and federal agencies is essential in modern healthcare systems to improve healthcare standards. However, transferring patient information between these entities can be challenging due to regulations like the General Data Protection Regulation (GDPR) or the European Health Data Space (EHDS). Through DFL, each hospital or institute only needs to share the parameters of the local model to obtain an accurate diagnosis model, thereby preserving patients’ privacy. Notably, the successful application of DFL-based approaches in the healthcare sector is evident from recent studies, as demonstrated in the relevant articles presented in Table V.

The first study to present the privacy issues of server use in healthcare is [160]. The authors noted the challenges in achieving a trade-off between global learning of the model and local information from each data source due to the large number of data sources with varying amounts and properties of data. To address this, they proposed a new strategy called Federated-Autonomous Deep Learning (FADL), where the first layer of the NN model is trained in a federated way using data from all sources. In contrast, the other model layers are trained locally in each data source. Their approach demonstrated an accuracy similar to the centralized analysis and outperformed regular FL for distributed electronic health records.

In the same way as the previous work, Huang et al. [128] proposed a community-based FL algorithm to predict mortality and hospital stay time. Electronic medical records were clustered into communities inside each hospital based on standard medical aspects. Each cluster learned and shared a particular ML model customized for each community rather than a general global model shared among all hospitals. In this work, a participant with an aggregator role was in charge of converting patients’ drug features into privacy-preserving representations. The authors of [162] introduced an algorithm to perform local updates during several iterations, enabling communication between nodes and improving communication efficiency for DFL. This approach was tested on simulations with real-world electronic medical records, demonstrating its effectiveness in Alzheimer’s disease detection compared to classical methods. Similarly, Elayan et al. [163] proposed

TABLE V
COMPARISON OF DFL SOLUTIONS FOCUSED ON HEALTHCARE

Ref.	Year	Goal	Federation Architecture	Topology	Scenario	ML Model	Aggregation Algorithm	Local Model Aggregation	Client Selection	Dataset
[160]	2018	Improve the accuracy of the CFL approximation by training parts of the model in a decentralized way and others using specific data sources	Cross-silo	Centralized Decentralized	Simulation	Three-layered NN	Custom algorithm	Synchronous	Sequential	eICU-CRD [161]
[128]	2019	Decrease the cost of communications between hospitals and the aggregator by using a clinical community data clustering algorithm for disease detection	Cross-silo	Centralized Decentralized	Simulation	K-Means	FedAvg	Synchronous	Sequential	eICU-CRD [161]
[162]	2020	Improve the communication efficiency for fully DFL over a graph without loss of optimality of the solutions	Cross-device	Decentralized	Real	Custom model	Custom algorithm	Synchronous Asynchronous	Sequential	Private
[163]	2021	Deep FL framework for decentralized healthcare systems that maintain user privacy	Cross-device	Decentralized	Simulation	Custom model	Custom algorithm	Synchronous	Scheduling	Private
[164]	2021	Facilitate the integration of any medical data from any data owner worldwide without violating privacy laws	Cross-silo	Decentralized	Real	Custom model	Custom aggregator	Asynchronous	Scheduling	PBMC transcriptome [165] Chest X-ray [166]
[167]	2022	Secure valuable hospital biomedical data useful for clinical research organizations, use of Blockchain and smart contracts	Cross-device	Decentralized	Simulation	CNN	FedAVG	Asynchronous	Sequential	Private
[16]	2022	Privacy and protection of medical data exchanged between healthcare entities	Cross-silo	Decentralized	Real	ResNet18 ResNet50 DenseNet121	Custom algorithm	Synchronous	Sequential	ImageNet [168] Private
[98]	2022	Detect brain tumor segmentation, using representative clinical datasets	Cross-device	Centralized Decentralized	Real	U-Net	CFA FedPer	Synchronous Asynchronous	Random	BRATS [169] Private
[90]	2022	Detect misconduct model on specific sites from any learning iteration	Cross-device	Centralized Decentralized	Simulation	GloreChain	FedAvg	Asynchronous	Sequential	Cancer Biomarker [170]
[2]	2022	Collaboratively training between healthcare devices for creating a robust model without raw data exchange	Cross-device	Decentralized	Simulation	3D CNN	Custom FedAvg (RingAVG)	Synchronous Asynchronous	Random	COVID-19 CT [171] Eye Disease [172] Skin Cancer [173]
[21]	2022	Ring topology for DGM to ensure security and privacy in medical scenarios	Cross-device	Decentralized	Simulation	GAN	FedAvg	Asynchronous	Sequential	MNIST [174]
[175]	2022	Exploit concepts from experience repetition and adversarial generation research	Cross-device	Centralized Decentralized	Simulation	GAN	FedAvg FedProx FedBN	Synchronous Asynchronous	Random	X-ray data [176], [177] HAM10000 [178]
[179]	2023	Privacy-preserving decentralized approach while supporting efficient communications	Cross-silo	Decentralized	Simulation	N/S	Custom algorithm	Synchronous	Random	MNIST [174]

a framework to train on skin lesion images using smartphones. They further utilized TFL to overcome the need for large, labeled data. They also proposed an automated training data-acquiring process and evaluated the algorithm on skin diseases, leveraging TL techniques to address the problem of a lack of healthcare data in generating DL models. Moreover, Wernat-Herresthal et al. [164] followed a similar approach to ensure the privacy of medical data exchanged between cross-silos.

The advancement of clinical research and new scientific discoveries relies on enabling data mobility, including medical records. In this context, various schemes have been proposed to exchange medical records using Blockchain and smart contracts to ensure appropriate data privacy and protection. Salim and Park [167] compared the results of their decentralized Convolutional Neural Network (CNN) model for Electronic Health Records (EHR) with a private Interplanetary File System (IPFS) and found promising results in accuracy, sensitivity, and specificity, similar to the traditional centralized model. In contrast, Nguyen et al. [16] discussed the use of federated distillation to decrease model complexity, using a private dataset called The Noisy Blind consisting of 1198 original images collected from four clinics. Their solution provided superior performance to traditional centralized training when nodes comprise low-quality data, which is common in healthcare. Similarly, the authors of [98] focused on medical image prediction for brain tumor segmentation, enabling the collaboration of multiple institutions by sharing their local computational models and training a U-Net model. This approach reached a target Dice Similarity Coefficient (DSC) above 85% using private data from patients' devices (e.g., heart rate, blood oxygen saturation).

B. Industry 4.0

The use of federated and decentralized techniques has increased considerably in Industry 4.0 to overcome the lack of communication between devices and sensors without needing a central node. As shown in Table VI, several recent studies have shown promising results in applying DFL-based approaches in industrial settings.

In [8], the authors discussed the opportunities and applications of DFL in automated and networked industrial systems supported by Device-to-Device (D2D) communications. The authors demonstrated the integration of the DFL approach into the sensing-decision-action loop, improving knowledge discovery operations. The system evaluation showed a convergence of the collaboration model in cross-device environments with minimal synchronous and asynchronous communications. Following the same direction, the authors of [181] applied cognitive computing in a simulated industrial scenario. This technique models the reasoning process of the human brain, having a progressive welcome in Industry 4.0 automation. This work implemented a DFL scenario supported by Blockchain and a Distributed Approximate Newton (DANE) aggregation method with advanced checks and random node selection. The results of an extensive evaluation and assessment showed accuracy values between 0.74 and 0.82. Ma et al. [52] defined Q-DFL for a quantized selection of network nodes. The proposal contained two phases: (i) a local model was trained with the SGD algorithm on each IIoT device, and then the parameters of the quantized model were exchanged among neighboring nodes; and (ii) a consensus mechanism was designed to ensure that the local models converged to the same global model. Subsequent simulation for the MobileNet model revealed its performance trade-off between system information flow consumption, time delay, and energy cost.

TABLE VI
COMPARISON OF DFL SOLUTIONS FOCUSED ON INDUSTRY 4.0

Ref.	Year	Goal	Federation Architecture	Topology	Scenario	ML Model	Aggregation Algorithm	Local Model Aggregation	Client Selection	Dataset
[8]	2021	Explore emerging FL opportunities using next-generation networked and autonomous industrial systems (e.g., robots, vehicles, drones)	Cross-device	Centralized Decentralized	Real	DNN	FedAvg	Synchronous Asynchronous	Random	MIMO radar [180]
[181]	2021	Decentralized paradigm for big data-based cognitive computing	Cross-device	Decentralized	Simulation	CNN	DANE	Synchronous	Random	CIFAR-10 [154]
[52]	2021	A method for D2D network in industrial IoT devices for improving the convergence of communications	Cross-device	Decentralized	Real	MobileNet	FedAvg (consensus)	Asynchronous	Random	MNIST [174]
[126]	2021	Anomaly detection ML models in non-IID scenarios, including mechanisms to locally rebalance training datasets	Cross-device	Decentralized	Simulation	Four-layered FCNN	P2PK-SMOTE	Synchronous Asynchronous	Random	N-BaIoT [182]
[183]	2022	A method for decentralized anomaly detection using neural networks, offering a comparison with traditional federated architectures	Cross-device	Decentralized	Simulation	Custom model	FedAvg	Asynchronous	Sequential	IoT23 [184]
[185]	2022	A BoEI framework that integrates an innovative DFL for cyberattack detection in IIoT	Cross-device	Centralized Decentralized	Simulation	Custom model	Fed-Trust	Synchronous Asynchronous	Random	TON_IoT [186] LITNET-2020 [187]
[93]	2022	Secure application approach to seal and sign asynchronous and synchronous FL collaborative tasks	Cross-silo Cross-device	Decentralized	Real	Custom model	Custom algorithm	Asynchronous	Sequential	Private
[59]	2022	Generic decentralized FL framework for training distributed models in inherently heterogeneous IoT environments	Cross-device	Decentralized	Real	Custom model	Custom algorithm	Synchronous Asynchronous	Random	Private
[188]	2022	Cooperative machine learning organized by physically close nodes	Cross-device	Decentralized	Simulation	Two-layered FCNN	WAFL	Synchronous Asynchronous	Random	MNIST [174]
[17]	2022	User-defined privacy-preserving framework for industrial metaverses	Cross-silo Cross-device	Decentralized	Simulation	N/S	Custom FedAvg	Asynchronous	Scheduling	Private
[189]	2023	A method that leverages Blockchain to improve model quality and reduce latency	Cross-silo	Decentralized	Simulation	Two-layered MLP	FedAvg	Asynchronous	Random	MNIST [174]
[138]	2023	Cooperative use of Blockchain and FL while preserving privacy using 5G	Cross-silo	Decentralized	Simulation	N/S	Custom algorithm	Synchronous	Random	CIFAR-10 [154] FEMNIST [147]

N/S (Not Specified) by the authors

Cybersecurity is also an important consideration in Industry 4.0, and several studies have investigated the use of DFL for anomaly and attack detection, such as the work published in [126]. The authors developed anomaly detection ML models for non-IID scenarios and included mechanisms to locally rebalance the training datasets through the synthetic generation of data points from the minority class. Using different metrics to evaluate the model performance on the N-BaIoT dataset, they obtained a convergence close to 75% of False Negative Rate (FNR) and recall. Another study, conducted by Lian and Su [183], discussed a similar approach with the IoT23 dataset [184], obtaining a model convergence 30% faster than in the previous study with an increase of only 3% in the overhead of the participants.

Apart from ensuring the reliability of IIoT and the connectivity of industrial objects, Abdel-Basset et al. [185] proposed a Blockchain-orchestrated Edge Intelligence (BoEI) framework for cyberattack detection in IIoT. The framework uses a temporal convolutional generative network to enable semi-supervised learning from semi-labeled data. The authors used TON_IoT [186] and LITNET-2020 [187] to validate the robustness and efficiency of the framework over different cyberattack detection approaches. Several studies have also addressed the synchronous or asynchronous operation of nodes within the network topology. In this regard, Pinyoanuntapong et al. [59] implemented a solution to mitigate high communication congestion in decentralized networks, leading to robust distributed model training in heterogeneous IoT environments where stragglers (i.e., slow devices) are commonplace due to varying computation and network speeds of IoT devices. Similarly, Ochiai et al. [188] proposed a similar approach for ad-hoc wireless environments, achieving an average accuracy of 95% for the test IID dataset compared to on-device training, which obtained

84.7%. Mothukuri et al. [93] proposed a Blockchain-based approach and Hyperledger Fabric with a gamification component for integrating a secure application to seal and sign asynchronous and synchronous DFL collaborative tasks. Their results demonstrated that security enhancements improve the FL process using a custom aggregation algorithm and asynchronous communications. Finally, Kang et al. [17] enhanced privacy in the industrial metaverse by using a DFL framework with cross-chain power for decentralized, secure, and privacy-preserving data training in physical and virtual spaces. The framework uses a hierarchical Blockchain architecture with the main chain and multiple sub-chains. Ranathunga et al. [189] introduced a similar cross-silo architecture using a network of aggregators to optimize model quality, minimize convergence time, and improve system performance in various industrial applications. Similarly, Singh et al. [138] proposed an approach for Industry 5.0. This scheme allowed local learning updates within the different departments of the industry using 5G. The validation outcomes showed an accuracy of 93.5% in a 50% active node.

C. Mobile Services

The proliferation of interconnected mobile devices and the development of advanced intelligent models have facilitated the emergence of decentralized approaches that offer increased efficiency and optimal training of models involved in various everyday tasks. To this end, DFL-based approaches have gained significant attention in mobile services, as demonstrated by the literature in Table VII.

Most research in this field has focused on leveraging edge computing to bring processing and data storage closer to the source of the request, thereby improving response times and conserving bandwidth on constrained devices. In this context,

TABLE VII
COMPARISON OF DFL SOLUTIONS FOCUSED ON MOBILE SERVICES

Ref.	Year	Goal	Federation Architecture	Topology	Scenario	ML Model	Aggregation Algorithm	Local Model Aggregation	Client Selection	Dataset
[7]	2021	Reduce the number of directly connected end devices, avoiding the overhead of unnecessary local updates	Cross-silo Cross-device	Centralized Decentralized	Real	CNN MLP	FedAvg	Synchronous Asynchronous	Random	MNIST [174]
[190]	2021	Novel approach based on continuous authentication of users trained by an organized peer-to-peer federation involving different organizations	Cross-device	Centralized Decentralized	Simulation	Custom model	FedAvg	Asynchronous	Scheduling	Private
[191]	2021	Blockchain-based FL scheme to strengthen communication security and data privacy protection in DITENs	Cross-silo Cross-device	Centralized Decentralized	Simulation	DNN	Custom algorithm	Synchronous Asynchronous	Random	MNIST [174] FMNIST [192]
[193]	2022	Simulation tool to capture the decentralized and asynchronous nature of FL operation in conjunction with DTL techniques	Cross-device	Decentralized	Simulation	U-Net	FedAvg	Asynchronous	Sequential	BRATS [169] Private
[68]	2022	Reputation scheme to balance the weights between trust values of parameters and bid prices	Cross-device	Centralized Decentralized	Simulation	Four-layered FNN	FedAvg	Synchronous	Scheduling	MNIST [174]
[18]	2022	Novel distributed validation weighting scheme to evaluate the performance of a mobile node in the federation versus a distributed validation set	Cross-device	Centralized Decentralized	Simulation	Two-layered CNN	Custom algorithm	Synchronous Asynchronous	Sequential	CIFAR-10 [154] CIFAR-100 [154] EMNIST [194]
[72]	2022	Decentralized recommender system based on the principles of gossip learning	Cross-device	Decentralized	Simulation	GMF PRME-G	FedAvg FedFast Reptile	Synchronous	Scheduling	Foursquare-NYC [195] Gowalla-NYC [196] MovieLens [197]
[73]	2022	Gossip-based system to maximize social awareness among nodes, improving reliability and latency	Cross-device	Decentralized	Simulation	N/S	Custom algorithm	Asynchronous	Sequential	Facebook [198]
[199]	2023	Federated GNN supporting supervised and unsupervised learning with security measures	Cross-device	Centralized Decentralized	Simulation	GCN GAT	Custom algorithm	Synchronous	Sequential	Facebook [198] LastFM [200]
[201]	2023	Privacy-preserving P2P learning for asynchronous and collaborative tasks.	Cross-device	Centralized Decentralized	Simulation	MobileNetv2 BLSTM	Custom algorithm	Synchronous Asynchronous	Random	CIFAR-10 [154] Avito [202] IMDb [203]
[204]	2023	Personalized and fully decentralized FL algorithm using knowledge distillation	Cross-device	Decentralized	Simulation	Three-layered NN CNN	Custom algorithm	Synchronous Asynchronous	Random	IoT identification [205] EMNIST [194]
[206]	2023	Decentralized and privacy-preserving FL for improved performance and scalability	Cross-device	Decentralized	Simulation	CNN	FedAvg	Synchronous	Random	MNIST [174]

N/S (Not Specified) by the authors

Wang et al. [7] proposed a novel solution that deployed mobile edge nodes at various network locations to act as communication hubs between the cloud and end devices. This approach effectively sidesteps the latency associated with high server concurrency. Moreover, the proposed method filters unnecessary models and communications using cosine similarity. Experimental results showed that the proposed scheme reduces the number of local updates by 60% compared to CFL and increases the convergence speed of the evaluated model by 10.3%. Additionally, Monschein et al. [190] addressed the challenge of acquiring large amounts of data to train powerful ML models, which is often an overwhelming task for a single organization. The authors proposed a methodology combining the establishment of continuous user-based authentication with federated and decentralized data governance.

As in the previous scenarios, DLT is recurrently used to ensure communications in decentralized scenarios. In this sense, Lu et al. [191] conducted a theoretical analysis to enhance communication security and protect data privacy in Digital Twin Edge Networks (DITENs). The proposed scheme demonstrated significant improvements in communication efficiency and data security for IoT applications, as confirmed by numerical results. Wilhelmi et al. [193] studied the impact of ledger inconsistencies on DFL performance. The study employed a reliable simulation tool that captures the decentralized and asynchronous nature of Blockchain operation. Qi et al. [68] proposed a hybrid Blockchain-based incentive mechanism that addresses similar challenges. The authors leveraged smart contracts and dynamic reputation score calculation of each DFL participant. In contrast to previous studies, performance is evaluated against malicious or non-honest nodes. In edge computing settings where

P2P communication is commonly used for exchanging model parameters between nodes, an efficient algorithm named CoCo was introduced in [18]. This algorithm integrates topology construction optimization and model compression to accelerate DFL. Extensive simulation results show that CoCo achieves a ten-fold speedup and reduces the communication cost by 50% on average compared to existing DFL baselines.

The rise of decentralized scenarios has enabled new applications such as recommender systems and social networks. In particular, Belal et al. [72] introduced PEPPER, a decentralized recommender system based on gossip learning principles. The system extracts relevant content for users to aid them in their daily activities, such as finding relevant places to visit, content to consume, or items to buy. In PEPPER, users gossip about model updates and aggregate them asynchronously. By conducting experiments on three real datasets implementing two use cases, location registration recommendation and movie recommendation, the authors demonstrated that their solution converges up to 42% faster than other decentralized solutions. Furthermore, Khelghatdoust and Mahdavi [73] proposed a Decentralized Online Social Network (DOSN) based on gossip to address the privacy and scalability issues of centralized social networks. The authors observed almost a 30% reduction in search latency and a 10% improvement in communication reliability through trusted contacts. These results demonstrate the potential of decentralized approaches in providing efficient solutions for modern applications such as recommender systems and social networks. To improve the previous scenarios, Pan et al. [199] utilized a tree constructor to improve representation capability given the limited structural information and a Monte Carlo Markov Chain-based algorithm to mitigate workload imbalance caused by degree

TABLE VIII
COMPARISON OF DFL SOLUTIONS RELATED TO THE MILITARY SCENARIO

Ref.	Year	Goal	Federation Architecture	Topology	Scenario	ML Model	Aggregation Algorithm	Local Model Aggregation	Client Selection	Dataset
[208]	2020	Federated learning-based security architecture for jamming attack detection for FANET	Cross-device	Centralized Decentralized	Simulation	Three-layered NN	FedAvg	Asynchronous	Random	CRAWDAD [209] NS3 FANET [210]
[211]	2020	Distributed defense solution for sustainable society using the features of Blockchain technology and federated learning	Cross-device	Centralized Decentralized	Simulation	CNN	FedAvg	Synchronous	Random	Custom dataset
[51]	2021	DFL solution with a fully connected network between battlefield devices employing a method of random walks and alternate directions during federation	Cross-device	Decentralized	Real	ELM	ISPW-ADMM	Asynchronous	Random	Private
[212]	2021	Novel architecture that allows FL within UAV networks without a central entity	Cross-device	Decentralized	Simulation	CNN	FedAvg	Asynchronous	Random	N/S
[19]	2021	Secure framework for UAV-assisted mobile crowdsensing to promote high-quality model sharing	Cross-device	Decentralized	Simulation	Custom model	Custom FedAvg	Synchronous	Scheduling	N/S
[65]	2021	Precoding and decoding strategies for D2D communication	Cross-device	Decentralized	Real	Custom model	FedAvg	Synchronous	Scheduling	BRATS [169]
[213]	2022	Novel integration of Blockchain and FL in UAV edge computing networks and associated challenges and solutions	Cross-device	Decentralized	Simulation	N/S	Custom FedAvg	Asynchronous	Sequential	N/S
[42]	2022	Cross-domain authentication of UAVs using multi-signature smart contracts	Cross-device	Decentralized	Simulation	CNN	Custom algorithm	Synchronous	Random	EMNIST [194]
[214]	2023	Resource-efficient framework for mmWave aerial-terrestrial integrated networks using UAVs	Cross-device	Decentralized	Simulation	N/S	Custom algorithm	Asynchronous	Scheduling	MNIST [174] CIFAR-10 [154]
[215]	2023	Feasibility and adaptability of DFL in maritime transportation	Cross-device	Centralized Decentralized	Simulation	Fully-connected ANN	Custom algorithm	Synchronous	Random	Private

N/S (Not Specified) by the authors

heterogeneity. Finally, it is worth mentioning that other studies have explored the use of innovative techniques in DFL, such as P2P with knowledge distillation, for various applications such as social network scenarios or IoT device identification [201], [204], [206].

D. Military

The military scenario has also yielded numerous improvements in DFL communications between different devices on the battlefield. A clear example is the European Future Combat Air System (FCAS) program [207], the largest European armament effort since World War II, where unmanned aircraft belong to a decentralized and collaborative system during combat missions. Table VIII highlights the most relevant items in the military scenario where DFL-based approaches are applied.

Flying Ad-hoc Network (FANET) is a type of network that applies DFL as a promising way to train collaborative and decentralized models. One challenge in this network is the evasion of malicious attacks, such as jamming. In jamming attacks, adversaries disrupt the victim network communication by generating interference at the receiver side, making it difficult for the intended signals to be correctly received and processed. For this problem, Mowla et al. [208] proposed security measures to prevent node jamming by creating a client group prioritization technique leveraging the Dempster-Shafer theory and using asynchronous communications to exchange model parameters. The results provided an accuracy of 82.01% for the CRAWAD dataset and 89.73% for the NS3 FANET dataset. Additionally, Sharma et al. [211] discussed using DFL in the Internet of Battlefield Things (IoBT) as a defense system employing AI to strengthen the armed forces. The authors rely on the characteristics of Blockchain technology to obtain high accuracy and a random device selection. A custom dataset consisting of several drone detection datasets [216] and images [217] is used to evaluate the effectiveness of

the proposed model, obtaining an accuracy rate of 99% at the fog layer. Xiao et al. [51] presented an Inexact Parallel Random Walk Alternate Direction Multiplier Method (ISPW-ADMM) applied to fully connected networks. Then, a UAV communication scenario was deployed and compared with traditional methods such as W-ADMM, PW-ADMM, DGD, and distributed-ADMM (D-ADMM). Qu et al. [212] proposed a similar solution, where the authors provided three critical considerations: (i) the average loss achieved by DFL is similar to the one obtained in a centralized approach; (ii) for each UAV, the loss value after 60 rounds is similar to the achieved in the centralized approach; and (iii) the training latency is always more negligible in DFL since it does not need to broadcast the model parameters.

Alternative solutions have been proposed to address the issue of secure DFL for UAV-assisted Mobile Crowdsensing (MC). For instance, Wang et al. [19] introduced a DFL approach that utilizes Blockchain to securely exchange local model updates and verify contributions without a central node in a cross-device scenario. In the same way, Shi et al. [65] proposed a consensus phase based on additive noise at each iteration of the algorithm, which enhanced the robustness of the solution to changes in wireless network topology. The solution converged linearly in a binary classification using MNIST. Furthermore, Zhu et al. [213] used the same technology among multiple untrusted parties with anonymous, immutable, and distributed characteristics, achieving similar results in performance for UAV intelligent edge computing networks to the previous study. Another study by Feng et al. [42] followed a similar approach, utilizing HFL and non-IID data in a fully connected network. The authors implemented cross-domain UAV authentication through multi-signature smart contracts, with global model updates computed using these smart contracts instead of a centralized server. Recently, decentralized model dissemination has emerged as a promising approach for DFL in mmWave aerial-terrestrial integrated networks. To this end, Al-Abiad et al. [214] presented an algorithm that

TABLE IX
COMPARISON OF DFL SOLUTIONS DEALING WITH VEHICULAR SCENARIOS

Ref.	Year	Goal	Federation Architecture	Topology	Scenario	ML Model	Aggregation Algorithm	Local Model Aggregation	Client Selection	Dataset
[20]	2020	Proactive caching scheme where vehicles are trained from sparse data, mitigating privacy risks	Cross-device	Decentralized	Real	CF-VAE	FedAvg	Asynchronous	Random	MovieLens [197]
[219]	2020	Asynchronous FL-based approach for efficient and secure data sharing	Cross-device	Decentralized	Simulation	Custom NN	FedAvg	Asynchronous	Scheduling	N/S
[220]	2020	Autonomous Blockchain-based powered learning design for a vehicular communication network	Cross-device	Decentralized	Simulation	N/S	oVML	Synchronous Asynchronous	Random	N/S
[22]	2021	Novel method proposed by DFL with P2P communication between autonomous vehicles	Cross-device	Centralized Decentralized	Simulation	CNN	Custom algorithm	Synchronous	Schematic	MNIST [174] KITTI [221]
[3]	2022	DFL system to increase road user and object classification capability based on Lidar data	Cross-device	Centralized Decentralized	Simulation	PointNet ML	Custom FedAvg	Synchronous	Random	nuScenes [222]
[223]	2023	Reinforcement Learning with FL agent for predictive QoS in teleoperated driving scenarios	Cross-device	Decentralized	Simulation	DNN	Custom algorithm	Asynchronous	Random	KITTI [221]
[224]	2023	Blockchain-enhanced DFL for efficient and privacy-protective data sharing	Cross-device	Decentralized	Simulation	CNN	FedAvg	Asynchronous	Scheduling	FMNIST [192]

N/S (Not Specified) by the authors

makes use of UAVs as local model aggregators through UAV-to-UAV communications and reduces the energy consumption of DFL using Radio Resource Management (RRM) under the constraints of over-the-air learning latency. In maritime environments, optimizing transportation focusing on certain performance metrics may lead to non-convex problems due to the large number and heterogeneity of network nodes and vessels. To tackle this issue, Giannopoulos et al. [215] presented and analyzed various use cases in these scenarios and demonstrated the superiority of DFL over traditional ML approaches using datasets from an enterprise specializing in the maritime industry.

E. Vehicles

In recent years, concerns regarding road safety have significantly increased, prompting significant efforts towards developing automated solutions to detect distractions while driving and to alert vehicles on the road intelligently [218]. In this sense, decentralized solutions and FL have emerged as promising approaches in this sector, as illustrated by the latest vehicle solutions highlighted in Table IX.

The unique characteristics of Vehicle-to-Everything (V2X) communication, including the high mobility of vehicles and limited storage capacity of nodes, present particular communication and processing challenges that can be addressed through DFL. Building on recent advances in ML, Yu et al. [20] proposed a proactive caching scheme based on SDFL and P2P communications. In this solution, a vehicle acts as a parameter server to aggregate the updated global model from peers instead of an edge node. Experimental results show that the solution outperforms typical baselines, gaining efficiency and autonomy. Furthermore, Blockchain supports data integrity when transmitting model parameters, thus addressing privacy concerns when sharing private data between vehicles. This mechanism has been utilized in several studies, such as [219] and [220]. While the former used deep Reinforcement Learning (RL) to select the participating nodes of DFL, thereby improving the efficiency of the data-sharing process, the latter minimized the system delay by exploiting the channel dynamics.

Finally, other studies consider new ways to optimize resources to present a solution comparable to server-based architectures. Chen et al. [22] proposed a novel approach to DFL based on a P2P network designed to be resilient to byzantine faults. The authors evaluated their method on the MNIST and KITTI datasets, assessing the performance of the proposed P2P approach in terms of accuracy, convergence speed, and fault tolerance and comparing it to traditional server-based methods. Similarly, Barbieri et al. [3] investigated the use of DFL methods to improve the classification of road users (such as pedestrians, cyclists, and vehicles) and objects based on Lidar data. The authors simulated a realistic V2X network using the collective perception service to share PointNet model parameters. The results showed a low latency training compared to existing distributed ML approaches. Another study that investigated the use of DFL in vehicular networks is presented in [223]. In this work, the authors focused on ensuring the Quality of Service (QoS) for communication between vehicles and remote drivers in remote driving scenarios. Specifically, they proposed using Predictive Quality of Service (PQoS) to predict and react to unanticipated degradation of the QoS. To implement PQoS in vehicular networks, the authors designed an RL agent to identify the optimal compression level for sending automotive data under low latency and reliability constraints. Finally, Hu et al. [224] integrated Blockchain into a fully connected vehicle topology. The authors demonstrated robust privacy protection and system reliability without significant latency, ensuring efficient and accurate data exchange, even with data of varying quality.

VI. TRENDS, LESSONS LEARNED, AND OPEN CHALLENGES

Based on the different aspects of DFL analyzed through research questions Q1-Q3, and following the storyline presented in Figure 2, this section responds to “Q4. What trends, lessons learned, and challenges have emerged in DFL?” To provide a clear structure, each trend, lesson, and challenge is related to a specific category, such as [Fund.] for fundamentals, [Fram.] for frameworks, and [Sec.] for application scenarios.

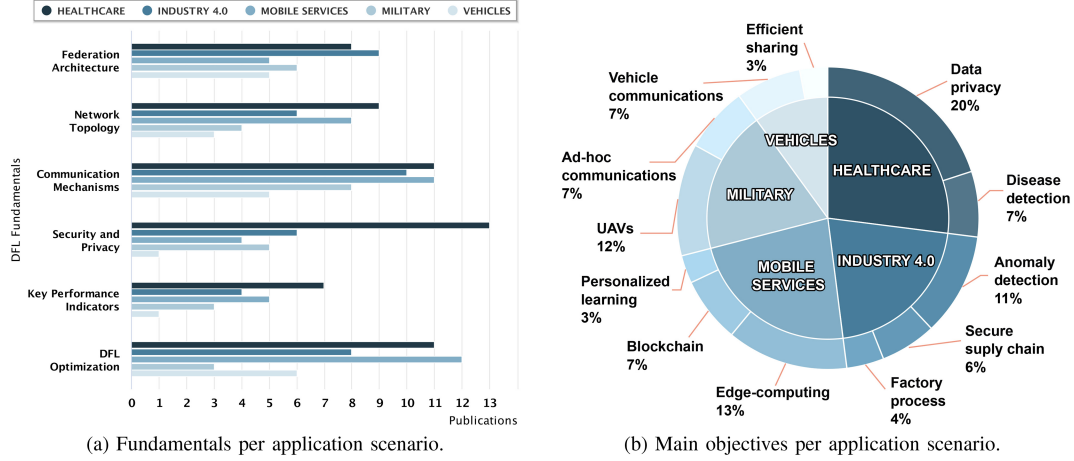


Fig. 9. Distribution of DFL publications.

A. Current Trends

The main trends in current work, based on the evolution of recent solutions, are as follows (see Figure 9).

[Fund.] *Federation architectures, network topologies, and communication mechanisms are extensively studied:* Most works analyze the communications between network nodes, the network topologies (mostly fully connected), and the architecture it supports, with DFL being the main approach addressed. Figure 9(a) shows the trend of the above fundamentals in different application scenarios.

[Fund.] *Fully connected network topologies are widely applied to DFL scenarios:* Network topologies where all nodes are connected to each other are the most commonly used in DFL. Its versatility and simplicity of construction have made it a usable approach in most application scenarios. Table II shows how about 50% of the papers analyzed in the network topology fundamental belong to this approach.

[Fund.] *The optimization of communications is predominant in recent work on DFL:* The inherent functionality of DFL requires communication optimization techniques between federation participants. In this sense, Table II shows how more than 65% of the analyzed solutions address complexity reduction in model parameter exchanges. The healthcare and mobile services are the most optimized application scenarios in terms of communications (see Figure 9(a)).

[Fram.] *Most of the frameworks are adapted to cross-device environments:* The reviewed frameworks offer DFL for deploying simulated or real federated scenarios with limited devices (see Table IV). They also typically provide models fed by images or timelines, aggregation algorithms, and various communication technologies between participants.

[Sce.] *DFL is widely used in healthcare scenarios:* DFL meets its goals in healthcare data analytics, with a significant impact on the accuracy of medicine and, ultimately, improved treatment and diagnosis (see Figure 9(b)). Electronic medical records, medical imaging, disease detection, and collaborative drug discovery are the most significant use cases for DFL.

[Sce.] *DFL gains momentum in mobile services and Industry 4.0:* The high performance offered by DFL has led to the increasing relevance of both application scenarios. In mobile services, DFL improves personalization and recommendation systems in edge devices while protecting user privacy. In Industry 4.0, DFL enables collaboration between multiple organizations without sharing sensitive data, reducing costs, and improving efficiency.

[Sce.] *Anomaly detection and deployment on UAVs are prominent trends:* Figure 9(b) shows how both topics maintain remarkable applicability in different application scenarios. Specifically, anomaly detection represents the 11% of works documented in industrial scenarios, while the use of UAVs represents the 12% of works in military scenarios.

B. Lessons Learned

After reviewing and analyzing the state of the art, the following lessons have been learned:

[Fund.] *The use of specific aggregation algorithms for DFL is still limited:* Although FedAvg is widely used in existing frameworks, applications in certain DFL scenarios need to obtain similar results. Studies of different application scenarios (see Table V, VI, VII, VIII, and IX) have opted for customizations of the algorithm to adapt it to the peculiarities of the federation models.

[Fund.] *Limited analysis of improving decentralized systems with DFL:* Decentralized systems face challenges such as consistency and coordination between nodes. In the literature, there are not enough studies comparing and evaluating DFL enhancement in these systems. There is a need to determine the resilience, robustness, and overall security provided by this approach that reduces dependency on a server.

[Fram.] *There is a limited number of solutions providing realistic federation benchmarks:* As more DFL architectures are developed, it is important to have a benchmark with representative datasets and workloads to evaluate existing systems and direct future development. Although many DFL

frameworks provide benchmarks (see Table IV), no single benchmark has been widely used in all studies reviewed. In addition, existing benchmarks often ignore metrics such as system efficiency, reliability, or architecture robustness. The evaluation of model performance on non-IID datasets and system security needs further research.

[Fram.] *There is no consensus on frameworks in the literature for deploying DFL architectures:* The frameworks analyzed in Table IV show heterogeneous characteristics, mostly adapted to validation scenarios. Therefore, there are limited open-source DFL frameworks with sufficient maturity to be network, node, and data agnostic.

[Sce.] *The military and vehicular scenarios are complex application scenarios for deploying DFL solutions:* The solutions presented in the military (see Table VIII) and vehicles scenarios (see Table IX) lack the robustness to be put into practice, so they usually recreate simulated deployments to adapt nodes and models. In addition, they present limited bandwidth, unstable network connections, and high-security requirements in simulations. Therefore, strategies such as edge computing, hybrid approaches, and Blockchain technology are used to reduce complexity.

[Sce.] *There is a lack of literature using unsupervised learning in DFL architectures:* The literature has not addressed the applicability of unsupervised learning in DFL-based scenarios where federation participants do not operate locally labeled data. Table VI, VII, VIII, and IX indicate that the literature only defines solutions using supervised ML models such as MLP or CNN for classification tasks.

C. Open Challenges

Based on the current state of the art, the following points represent the main challenges that future DFL solutions might consider. In addition, Table X summarizes the open challenges and their future developments, indicating the importance of their application in the future.

[Fund.] *Improve the scalability of the solution when the number of participants in the federation increases:* The development of algorithms that can dynamically select participants based on their availability, network connection, and trustworthiness would help to ensure that the federation remains resilient to dropouts and the unavailability of participants. In addition, exploring new techniques for compressing and aggregating models across multiple participants can reduce communication and computation overhead, particularly for large-scale federations. Finally, including personalized local model learning can improve training while minimizing the impact of low network dependability and client availability.

[Fund.] *Improve the cybersecurity mechanisms depending on the participant and the application scenario:* The security of the nodes participating in DFL and their ability to detect and prevent attacks or threats in heterogeneous scenarios are crucial to the success of this approach. A promising solution would be to design an architecture that treats participants differently according to their privacy restrictions, allowing for a more personalized and secure approach. This is particularly

TABLE X
OPEN CHALLENGES IN DFL AND FUTURE DEVELOPMENTS

Challenge	Future Developments
Fundamentals [Fund.]	
Scalability of DFL with increasing participants (!!!)	• Dynamic participant selection • Personalized local model learning
Cybersecurity mechanisms for a secure DFL (!!!)	• Detect attacks in DFL scenarios • Different treatment based on privacy
Trustworthiness among federation participants (!!!)	• Maintain trust policies • Prevent dishonest behavior
Homogeneous node participation (!)	• Quantization and gradient compression • Use of SDFL
Address participant mobility in DFL scenarios (!)	• Topology-aware node reconfiguration • Resilient synchronization methods
Study of adversarial attacks (!)	• Identify the techniques and their impacts • Compare against traditional approaches
Explore the use of Reinforcement Learning (!)	• Optimize the federated model performance • Improve the selection of participants
DFL standardization efforts (!)	• Promote comprehensive DFL standards • Involve standard-setting bodies (ISO, IEEE)
5G and 6G technologies for communications (!)	• Network slicing utilization • 5G/6G-integrated edge computing
Frameworks [Fram.]	
Modular, scalable, and efficient frameworks (!!!)	• Implement and manage DFL fundamentals • Application in practical scenarios
Heterogeneous datasets in decentralized participants (!)	• Data preprocessing and normalization • Advanced data augmentation techniques
Dynamic scheduling of federated network (!)	• Adaptable federation architecture • Resilient algorithms
Application scenarios [Sce.]	
Exploration of new DFL application scenarios (!)	• Evaluate DFL in smart city technologies • Combine AI, IoT, and DFL for testing tools

! low importance, !! high importance, !!! critical

relevant for application scenarios such as social media applications, industrial IoT, and Blockchain, where user privacy protection is critical.

[Fund.] *Enhance trustworthiness among federation participants in DFL approaches:* To ensure the accuracy and reliability of the federated model, participating nodes must establish trust policies with one another. These policies allow participants to aggregate model parameters based on reputation and performance history only from those they trust. By doing so, participants can avoid dishonest behavior and maintain the integrity of the FL process, thereby encouraging the participation of other nodes in the network.

[Fund.] *Ensure the homogeneous participation of the constrained nodes in the federation:* It is challenging to guarantee the participation of nodes with limited autonomy or bandwidth. Some solutions could use quantization methods and gradient compression techniques to reduce communication overhead. Another option is to consider other federation architectures, such as SDFL, where nodes with limited resources can participate in the federation while still maintaining some level of centralization.

[Fund.] *Address participant mobility:* The dynamic nature of DFL, where participants may join or leave the federation at any time, poses challenges in maintaining learning stability and reliability. Future DFL solutions need to manage mobility and develop strategies to handle the continuous flow of participants while ensuring efficient learning.

[Fund.] *In-depth study of adversarial attacks applied in DFL approaches:* The literature on cyberattacks affecting DFL is limited, with relatively little attention given to adversarial attacks and their impact on DFL performance. Thus, evaluating their impact compared to other traditional approaches, such as CFL, can help understand the vulnerabilities of DFL

better and inform the development of more robust security measures.

[Fund.] Explore the use of RL: The literature regarding RL applied in decentralized collaborative scenarios is scarce. This learning approach can optimize the performance of the federated model by dynamically allocating resources and adjusting the learning process. Additionally, it can be used to optimize the selection of participants in the federation, as well as to dynamically adjust the communication and aggregation parameters based on the performance of the participants.

[Fund.] Research the use of 5G/6G networks to improve DFL communications: The rise of 5G and subsequent 6G networks presents a significant potential to enhance communications performance by providing faster and more reliable communication between nodes. Thus, exploring techniques for efficient data aggregation and transmission is necessary using ultra-low latency and high bandwidth capabilities. Moreover, researchers could examine the possibility of utilizing network slicing to allocate dedicated resources for DFL tasks and integrate edge computing within these networks. Furthermore, with 5G/6G-powered DFL, a deep focus on the security and integrity of P2P communications between edge devices becomes critical. Given the potential vulnerability of these networks to novel security threats, dedicated research on defense mechanisms is essential to safeguard secure and trusted P2P exchanges. To this end, it is necessary to deploy appropriate defense mechanisms, such as encryption, SMPC, or Blockchain technologies.

[Fram.] Create modular, scalable, and efficient frameworks for diverse application scenarios: The literature has not addressed an agnostic federation for different solutions in terms of DFL fundamentals. Therefore, it is necessary to design and implement a robust and reliable solution to generate realistic scenarios, considering elements such as model creation, data storage, framework usage, model parameter exchange, and model aggregation. The framework should focus on efficiency, considering communication overhead, computational resources, and storage requirements.

[Fram.] Handle heterogeneous datasets in decentralized participants: Participants may possess varying sizes, distributions, and quality data, impacting the model performance. Addressing the challenge of dividing or adapting datasets to account for these differences is crucial to achieving optimal performance in DFL scenarios. It involves creating methods for data preprocessing, normalization, and augmentation that can accommodate the distinct characteristics and variations inherent in the datasets of each participating node.

[Fram.] Adapt the dynamic scheduling of the federated network to the application scenario: Due to the instability of the application scenario, the number of participants may not be fixed during the learning process in DFL. Therefore, it is necessary to develop algorithms that can dynamically adjust the number of participants in the federation based on changes in inputs or outputs from the nodes. This would require building resilient algorithms that handle low network dependability and client availability. In addition, exploring the use of adaptive FL techniques that can adjust the level of centralization

or decentralization in response to changes in the application scenario could be beneficial.

[Sce.] Explore new DFL application scenarios: DFL can be effectively applied in a multitude of application scenarios. For instance, DFL can help to increase the intelligence of robotic and autonomous systems, making them more efficient and reliable. In smart city technologies, DFL can exchange local model parameters between devices, enhancing overall performance. Additionally, DFL can be evaluated using Digital Twin (DT) technology, combining AI, IoT, and DFL to create a simulated environment where tools can be tested in real time without exposing people or facilities to unknown risks.

[Fund.] Explore standardization activities for DFL: It is necessary to have comprehensive standards to ensure uniformity, interoperability, and trustworthiness across various implementations. These standards should address key elements like privacy-preserving methods, security features, data quality standards, model training processes, and performance indicators. Standardizing DFL could encourage consistency in research outcomes and establish a universal framework for assessing DFL methodologies.

VII. CONCLUSION AND FUTURE WORK

This work studies the evolution of DFL in recent years, providing the basic and distinctive fundamentals versus traditional federated architectures, its current application scenarios, and the frameworks that manage the deployment of DFL architectures. In this context, the present work has answered the following research questions.

Q1. What are the fundamental aspects of DFL? Section III studies the main aspects of DFL in the most current and representative works in the literature, clarifying the differences with CFL. The analysis provides a set of fundamentals related to federation architectures, network topologies, communication mechanisms, and security techniques. It also proposes KPIs to evaluate the approach and mechanisms to optimize DFL. With all this, the work proposes the first taxonomy that defines and details the fundamentals of DFL.

Q2. What DFL frameworks exist, and what fundamentals do they provide? Section IV discusses the main frameworks currently used to deploy DFL scenarios. This section shows the most mature frameworks in FL versus the incipient solutions that favor the inclusion of DFL. Thus, this paper highlights the prominence of current solutions and the DFL fundamentals contemplated to generate robust solutions to the scenario.

Q3. Which are the main characteristics of the most relevant scenarios of DFL? Section V describes, analyzes, and compares the most relevant and recent solutions according to their application scenario. The predominant application scenarios are healthcare, mobile services, and Industry 4.0. Also, this work determines that fields such as military or vehicles have grown significantly in recent years. Regarding fundamentals, decentralized cross-device architectures in fully connected network topologies are predominant. Optimization mechanisms are applied to the above aspects, mainly to communications and aggregation algorithms.

Q4. What trends, lessons learned, and challenges have emerged in DFL? Lessons learned, current trends, and future challenges have been documented in Section VI. It details how to create more sophisticated federation architectures, topologies, optimizations, and their applicability in different scenarios. At the same time, certain limitations in the recent literature are also detailed. For instance, the scarce comparison between DFL scenarios or the lack of frameworks allowing their deployment. In addition, handling heterogeneous datasets, cyberattacks, or using 5G/6G in communications have not yet been robustly applied, opening avenues for future research.

As future work, it is planned to design and implement scalable solutions capable of generating heterogeneous scenarios. To achieve this, it is necessary to create solutions using agnostic data types employing performance data from the devices acting as nodes. These solutions will be integrated with ML/DL techniques, communication optimization, and data aggregation techniques that preserve security and privacy while guaranteeing network and node performance capabilities. Furthermore, these solutions must offer a range of configurable options based on various federation architectures, including CFL, DFL, or SDFL, as well as decentralized technologies such as Blockchain. Finally, it is considered to define and build sets of metrics to complement the functionality of the solution and the evaluation of the system.

REFERENCES

- [1] H. Liu and W. Wu, "Federated reinforcement learning for decentralized voltage control in distribution networks," *IEEE Trans. Smart Grid*, vol. 13, no. 5, pp. 3840–3843, Sep. 2022.
- [2] Z. Lian et al., "DEEP-FEL: Decentralized, efficient and privacy-enhanced federated edge learning for healthcare cyber physical systems," *IEEE Trans. Netw. Sci. Eng.*, vol. 9, no. 5, pp. 3558–3569, Sep/Oct. 2022.
- [3] L. Barbieri, S. Savazzi, M. Brambilla, and M. Nicoli, "Decentralized federated learning for extended sensing in 6G connected vehicles," *Veh. Commun.*, vol. 33, Jan. 2022, Art. no. 100396.
- [4] H. B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. Y. Arcas, "Communication-efficient learning of deep networks from decentralized data," 2016, *arXiv:1602.05629*.
- [5] L. He, A. Bian, and M. Jaggi, "COLA: Decentralized linear learning," in *Proc. 32nd Int. Conf. Neural Inf. Process. Syst. (NIPS)*, 2018, pp. 4541–4551.
- [6] A. Hard, K. Partridge, R. Mathews, and S. Augenstein, "Jointly learning from decentralized (federated) and centralized data to mitigate distribution shift," in *Proc. NeurIPS Workshop Distrib. Shifts*, 2021, pp. 1–8.
- [7] T. Wang, Y. Liu, X. Zheng, H.-N. Dai, W. Jia, and M. Xie, "Edge-based communication optimization for distributed federated learning," *IEEE Trans. Netw. Sci. Eng.*, vol. 9, no. 4, pp. 2015–2024, Jul./Aug. 2022.
- [8] S. Savazzi, M. Nicoli, M. Bennis, S. Kianoush, and L. Barbieri, "Opportunities of federated learning in connected, cooperative, and automated industrial systems," *IEEE Commun. Mag.*, vol. 59, no. 2, pp. 16–21, Feb. 2021.
- [9] A. Bellet, A.-M. Kermarrec, and E. Lavoie, "D-Cliques: Compensating for data heterogeneity with topology in decentralized federated learning," in *Proc. 41st Int. Symp. Rel. Distrib. Syst.*, 2022, p. 11. [Online]. Available: <http://infoscience.epfl.ch/record/294834>
- [10] E. Georgatos, C. Mavroukafalidis, and K. Berberidis, "Efficient fully distributed federated learning with adaptive local links," 2022, *arXiv:2203.12281*.
- [11] K. Bonawitz et al., "Towards federated learning at scale: System design," in *Proc. Mach. Learn. Syst.*, vol. 1, 2019, pp. 374–388.
- [12] Y. Qu, C. Xu, L. Gao, Y. Xiang, and S. Yu, "FL-SEC: Privacy-preserving decentralized federated learning using SignSGD for the Internet of artificially intelligent things," *IEEE Internet Things Mag.*, vol. 5, no. 1, pp. 85–90, Mar. 2022.
- [13] Z. Zhao, J. Xia, L. Fan, X. Lei, G. K. Karagiannis, and A. Nallanathan, "System optimization of federated learning networks with a constrained latency," *IEEE Trans. Veh. Technol.*, vol. 71, no. 1, pp. 1095–1100, Jan. 2022.
- [14] B. Wang, J. Fang, H. Li, X. Yuan, and Q. Ling, "Confederated learning: Federated learning with decentralized edge servers," *IEEE Trans. Signal Process.*, vol. 71, pp. 248–263, 2023, doi: [10.1109/TSP.2023.3241768](https://doi.org/10.1109/TSP.2023.3241768).
- [15] EPFL—Machine Learning and Optimization Laboratory. "DISCO—Distributed collaborative machine learning." 2022. [Online]. Available: <https://github.com/epflml/disco>
- [16] T. V. Nguyen et al., "A novel decentralized federated learning approach to train on globally distributed, poor quality, and protected private medical data," *Sci. Rep.*, vol. 12, no. 1, p. 8888, 2022.
- [17] J. Kang et al., "Blockchain-based federated learning for industrial meta-verses: Incentive scheme with optimal Aol," in *Proc. IEEE Int. Conf. Blockchain (Blockchain)*, 2022, pp. 71–78.
- [18] L. Wang, Y. Xu, H. Xu, M. Chen, and L. Huang, "Accelerating decentralized federated learning in heterogeneous edge computing," *IEEE Trans. Mobile Comput.*, vol. 22, no. 9, pp. 5001–5016, Sep. 2013.
- [19] Y. Wang, Z. Su, N. Zhang, and A. Benslimane, "Learning in the air: Secure federated learning for UAV-assisted crowdsensing," *IEEE Trans. Netw. Sci. Eng.*, vol. 8, no. 2, pp. 1055–1069, Apr.–Jun. 2021.
- [20] Z. Yu, J. Hu, G. Min, H. Xu, and J. Mills, "Proactive content caching for Internet-of-Vehicles based on peer-to-peer federated learning," in *Proc. IEEE 26th Int. Conf. Parallel Distrib. Syst.*, 2020, pp. 601–608.
- [21] Z. Wang, Y. Hu, S. Yan, Z. Wang, R. Hou, and C. Wu, "Efficient ring-topology decentralized federated learning with deep generative models for medical data in healthcare systems," *Electronics*, vol. 11, no. 10, p. 1548, 2022.
- [22] J.-H. Chen, M.-R. Chen, G.-Q. Zeng, and J.-S. Weng, "BDFL: A Byzantine-fault-tolerance decentralized federated learning method for autonomous vehicle," *IEEE Trans. Veh. Technol.*, vol. 70, no. 9, pp. 8639–8652, Sep. 2021.
- [23] W. Y. B. Lim et al., "Federated learning in mobile edge networks: A comprehensive survey," *IEEE Commun. Surveys Tuts.*, vol. 22, no. 3, pp. 2031–2063, 3rd Quart., 2020.
- [24] D. C. Nguyen, M. Ding, P. N. Pathirana, A. Seneviratne, J. Li, and H. V. Poor, "Federated learning for Internet of Things: A comprehensive survey," *IEEE Commun. Surveys Tuts.*, vol. 23, no. 3, pp. 1622–1658, 3rd Quart., 2021.
- [25] L. U. Khan, W. Saad, Z. Han, E. Hossain, and C. S. Hong, "Federated learning for Internet of Things: Recent advances, taxonomy, and open challenges," *IEEE Commun. Surveys Tuts.*, vol. 23, no. 3, pp. 1759–1799, 3rd Quart., 2021.
- [26] V. Mothukuri, R. M. Parizi, S. Pouriyeh, Y. Huang, A. Dehghantanha, and G. Srivastava, "A survey on security and privacy of federated learning," *Future Gener. Comput. Syst.*, vol. 115, pp. 619–640, Feb. 2021.
- [27] P. Boobalan et al., "Fusion of federated learning and Industrial Internet of Things: A survey," *Comput. Netw.*, vol. 212, Jul. 2022, Art. no. 109048.
- [28] M. Joshi, A. Pal, and M. Sankarasubbu, "Federated learning for healthcare domain—Pipeline, applications and challenges," *ACM Trans. Comput. Healthcare*, vol. 3, no. 4, pp. 1–36, 2022.
- [29] L. Witt, M. Heyer, K. Toyoda, W. Samek, and D. Li, "Decentral and incentivized federated learning frameworks: A systematic literature review," 2022, *arXiv:2205.07855*.
- [30] Y. Qu, M. P. Uddin, C. Gan, Y. Xiang, L. Gao, and J. Yearwood, "Blockchain-enabled federated learning: A survey," *ACM Comput. Surveys*, vol. 55, no. 4, pp. 1–35, 2022.
- [31] M. Billah, S. T. Mehedi, A. Anwar, Z. Rahman, and R. Islam, "A systematic literature review on blockchain enabled federated learning framework for Internet of Vehicles," 2022, *arXiv:2203.05192*.
- [32] R. Gupta and T. Alam, "Survey on federated-learning approaches in distributed environment," *Wireless Pers. Commun.*, vol. 125, pp. 1631–1652, Mar. 2022.
- [33] D. Saraswat et al., "Blockchain-based federated learning in UAVs beyond 5G networks: A solution taxonomy and future directions," *IEEE Access*, vol. 10, pp. 33154–33182, 2022.
- [34] J. Wu, S. Drew, F. Dong, Z. Zhu, and J. Zhou, "Topology-aware federated learning in edge computing: A comprehensive survey," 2023, *arXiv:2302.02573*.

- [35] H. Chen, H. Wang, D. Jin, and Y. Li, "Advancements in federated learning: Models, methods, and privacy," 2023, *arXiv:2302.11466*.
- [36] L. Witt, M. Heyer, K. Toyoda, W. Samek, and D. Li, "Decentral and incentivized federated learning frameworks: A systematic literature review," *IEEE Internet Things J.*, vol. 10, no. 4, pp. 3642–3663, Feb. 2023.
- [37] T. Li, A. K. Sahu, A. Talwalkar, and V. Smith, "Federated learning: Challenges, methods, and future directions," *IEEE Signal Process. Mag.*, vol. 37, no. 3, pp. 50–60, May 2020.
- [38] R. Dai, L. Shen, F. He, X. Tian, and D. Tao, "DisPFL: Towards communication-efficient personalized federated learning via decentralized sparse training," 2022, *arXiv:2206.00187*.
- [39] O. Marfoq, C. Xu, G. Neglia, and R. Vidal, "Throughput-optimal topology design for cross-silo federated learning," in *Proc. 34th Int. Conf. Neural Inf. Process. Syst. (NIPS)*, 2020, p. 4.
- [40] J. Han, Y. Han, G. Huang, and Y. Ma, "DeFL: Decentralized weight aggregation for cross-silo federated learning," 2022, *arXiv:2208.00848*.
- [41] S. P. Karimireddy et al., "Breaking the centralized barrier for cross-device federated learning," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 34, 2021, pp. 28663–28676.
- [42] C. Feng, B. Liu, K. Yu, S. K. Goudos, and S. Wan, "Blockchain-empowered decentralized horizontal federated learning for 5G-enabled UAVs," *IEEE Trans. Ind. Informat.*, vol. 18, no. 5, pp. 3582–3592, May 2022.
- [43] M. Wilt, J. K. Matelsky, and A. S. Gearhart, "Scatterbrained: A flexible and expandable pattern for decentralized machine learning," 2021, *arXiv:2112.07718*.
- [44] S. Kalra, J. Wen, J. C. Cresswell, M. Volkovs, and H. R. Tizhoosh, "ProxyFL: Decentralized federated learning through proxy model sharing," 2021, *arXiv:2111.11343*.
- [45] R. Kanagavelu et al., "Ce-Fed: Communication efficient multi-party computation enabled federated learning," *Array*, vol. 15, Sep. 2022, Art. no. 100207.
- [46] R. Xu and Y. Chen, "μDFL: A secure microchained decentralized federated learning fabric atop IoT networks," *IEEE Trans. Netw. Service Manag.*, vol. 19, no. 3, pp. 2677–2688, Sep. 2022.
- [47] M. Yemini, R. Saha, E. Ozfatura, D. Gündüz, and A. J. Goldsmith, "Semi-decentralized federated learning with collaborative relaying," in *Proc. IEEE Int. Symp. Inf. Theory*, 2022, pp. 1471–1476.
- [48] F. P.-C. Lin, S. Hosseinalipour, S. S. Azam, C. G. Brinton, and N. Michelusi, "Semi-decentralized federated learning with cooperative D2D local model aggregations," *IEEE J. Sel. Areas Commun.*, vol. 39, no. 12, pp. 3851–3869, Dec. 2021.
- [49] P. M. S. Sánchez et al., "Analyzing the robustness of decentralized horizontal and vertical federated learning architectures in a non-IID scenario," 2022, *arXiv:2210.11061*.
- [50] C. Li, G. Li, and P. K. Varshney, "Decentralized federated learning via mutual knowledge transfer," *IEEE Internet Things J.*, vol. 9, no. 2, pp. 1136–1147, Jan. 2022.
- [51] Y. Xiao et al., "Fully decentralized federated learning-based on-board mission for UAV swarm system," *IEEE Commun. Lett.*, vol. 25, no. 10, pp. 3296–3300, Oct. 2021.
- [52] T. Ma, H. Wang, and C. Li, "Quantized distributed federated learning for Industrial Internet of Things," *IEEE Internet Things J.*, vol. 10, no. 4, pp. 3027–3036, Feb. 2023.
- [53] Y.-T. Chow, W. Shi, T. Wu, and W. Yin, "Expander graph and communication-efficient decentralized optimization," in *Proc. 50th Asilomar Conf. Signals Syst. Comput.*, 2016, pp. 1715–1720.
- [54] Y. Hua, K. Miller, A. L. Bertozzi, C. Qian, and B. Wang, "Efficient and reliable overlay networks for decentralized federated learning," *SIAM J. Appl. Math.*, vol. 82, no. 4, pp. 1558–1586, 2022.
- [55] T. Vogels, H. Hendrikx, and M. Jaggi, "Beyond spectral gap: The role of the topology in decentralized learning," 2022, *arXiv:2206.03093*.
- [56] K. Cheng et al., "SecureBoost: A lossless federated learning framework," *IEEE Intell. Syst.*, vol. 36, no. 6, pp. 87–98, Nov/Dec. 2021.
- [57] C. Briggs, Z. Fan, and P. Andras, "Federated learning with hierarchical clustering of local updates to improve training on non-IID data," in *Proc. Int. Joint Conf. Neural Netw.*, 2020, pp. 1–9.
- [58] M. S. Al-Abiad, M. Obeed, M. J. Hossain, and A. Chaaban, "Decentralized aggregation for energy-efficient federated learning via d2d communications," *IEEE Trans. Commun.*, vol. 71, no. 6, pp. 3333–3351, Jun. 2023.
- [59] P. Pinyoanuntapong, W. H. Huff, M. Lee, C. Chen, and P. Wang, "Toward scalable and robust AIoT via decentralized federated learning," *IEEE Internet Things Mag.*, vol. 5, no. 1, pp. 30–35, Mar. 2022.
- [60] S. Zehtabi, S. Hosseinalipour, and C. G. Brinton, "Decentralized event-triggered federated learning with heterogeneous communication thresholds," 2022, *arXiv:2204.03726*.
- [61] J. Cao, Z. Lian, W. Liu, Z. Zhu, and C. Ji, "HADFL: Heterogeneity-aware decentralized federated learning framework," in *Proc. 58th ACM/IEEE Design Autom. Conf. (DAC)*, 2021, pp. 1–6.
- [62] Z. Chen, W. Liao, P. Tian, Q. Wang, and W. Yu, "A fairness-aware peer-to-peer decentralized learning framework with heterogeneous devices," *Future Internet*, vol. 14, no. 5, p. 138, 2022.
- [63] Y. Zhou, Q. Chen, Z. Wang, D. Xiao, and J. Chen, "Communication-efficient cluster federated learning in large-scale peer-to-peer networks," 2022, *arXiv:2204.03843*.
- [64] X. Wang, A. Laliitha, T. Javidi, and F. Koushanfar, "Peer-to-peer variational federated learning over arbitrary graphs," *IEEE J. Sel. Areas Inf. Theory*, vol. 3, no. 2, pp. 172–182, Jun. 2022.
- [65] Y. Shi, Y. Zhou, and Y. Shi, "Over-the-air decentralized federated learning," in *Proc. IEEE Int. Symp. Inf. Theory*, 2021, pp. 455–460.
- [66] Z. Li et al., "Mining latent relationships among clients: Peer-to-peer federated learning with adaptive neighbor matching," 2022, *arXiv:2203.12285*.
- [67] T. Wink and Z. Nochta, "An approach for peer-to-peer federated learning," in *Proc. 51st Annu. IEEE/FIP Int. Conf. Depend. Syst. Netw. Workshops*, 2021, pp. 150–157.
- [68] M. Qi, Z. Wang, S. Chen, and Y. Xiang, "A hybrid incentive mechanism for decentralized federated learning," *Distrib. Ledger Technol.*, vol. 1, no. 1, pp. 1–15, 2022.
- [69] I. Hegedus, G. Danner, and M. Jelasity, "Decentralized learning works: An empirical comparison of Gossip learning and federated learning," *J. Parallel Distrib. Comput.*, vol. 148, pp. 109–124, Feb. 2021.
- [70] Z. Tang, S. Shi, B. Li, and X. Chu, "GossipFL: A decentralized federated learning framework with sparsified and adaptive communication," *IEEE Trans. Parallel Distrib. Syst.*, vol. 34, no. 3, pp. 909–922, Mar. 2023.
- [71] J. Jiang, L. Hu, C. Hu, J. Liu, and Z. Wang, "BACombo: Bandwidth-aware decentralized federated learning," *Electronics*, vol. 9, no. 3, p. 440, 2020.
- [72] Y. Belal, A. Bellet, S. B. Mokhtar, and V. Nitu, "PEPPER: Empowering user-centric recommender systems over gossip learning," *Proc. ACM Interact. Mobile Wearable Ubiquitous Technol.*, vol. 6, no. 3, pp. 1–27, 2022.
- [73] M. Khelghatdoust and M. Mahdavi, "A socially-aware, privacy-preserving, and scalable federated learning protocol for distributed online social networks," in *Advanced Information Networking and Applications*, L. Barolli, F. Hussain, and T. Enokido, Eds. Cham, Switzerland: Springer Int., 2022, pp. 192–203.
- [74] A. Koloskova, S. U. Stich, and M. Jaggi, "Decentralized stochastic optimization and Gossip algorithms with compressed communication," in *Proc. 36th Int. Conf. Mach. Learn.*, vol. 97, 2019, pp. 3479–3487.
- [75] J. Yin, X. Cui, and K. Li, "A reputation-based resilient and recoverable P2P botnet," in *Proc. IEEE 2nd Int. Conf. Data Sci. Cybersec. (DSC)*, 2017, pp. 275–282.
- [76] R. S. Rawat, M. Diwakar, and P. Verma, "Zeroaccess botnet investigation and analysis," *Int. J. Inf. Technol.*, vol. 13, no. 5, pp. 2091–2099, 2021.
- [77] Y. Chen, F. Luo, T. Li, T. Xiang, Z. Liu, and J. Li, "A training-integrity privacy-preserving federated learning scheme with trusted execution environment," *Inf. Sci.*, vol. 522, pp. 69–79, Jun. 2020.
- [78] B. Alangot, D. Reijnders, S. Venugopalan, P. Szalachowski, and K. S. Yeo, "Decentralized and lightweight approach to detect eclipse attacks on proof of work blockchains," *IEEE Trans. Netw. Service Manag.*, vol. 18, no. 2, pp. 1659–1672, Jun. 2021.
- [79] B. Niu, Y. Chen, Z. Wang, F. Li, B. Wang, and H. Li, "Eclipse: Preserving differential location privacy against long-term observation attacks," *IEEE Trans. Mobile Comput.*, vol. 21, no. 1, pp. 125–138, Jan. 2022.
- [80] J. Verbracken, M. de Vos, and J. Pouwelse, "Bristle: Decentralized federated learning in Byzantine, non-I.I.D. environments," 2021, *arXiv:2110.11006*.
- [81] G. Lu, Z. Xiong, R. Li, and W. Li, "Decentralized federated learning: A defense against gradient inversion attack," in *Proc. Wireless Internet*, 2023, pp. 44–56.
- [82] S. Chen, D. Yu, Y. Zou, J. Yu, and X. Cheng, "Decentralized wireless federated learning with differential privacy," *IEEE Trans. Ind. Informat.*, vol. 18, no. 9, pp. 6273–6282, Sep. 2022.
- [83] E. T. M. Beltrán et al., "Mitigating communications threats in decentralized federated learning through moving target defense," 2023, *arXiv:2307.11730*.

- [84] Á. L. P. Gómez, E. T. M. Beltrán, P. M. S. Sánchez, and A. H. Celdrán, "TemporalFED: Detecting cyberattacks in industrial time-series data using decentralized federated learning," 2023, *arXiv:2308.03554*.
- [85] R. C. Geyer, T. Klein, and M. Nabi, "Differentially private federated learning: A client level perspective," 2017, *arXiv:1712.07557*.
- [86] B. Verhaelen, F. Mayer, S. Peukert, and G. Lanza, "A comprehensive KPI network for the performance measurement and management in global production networks," *Prod. Eng.*, vol. 15, no. 5, pp. 635–650, 2021.
- [87] Y. Qu et al., "Decentralized privacy using blockchain-enabled federated learning in fog computing," *IEEE Internet Things J.*, vol. 7, no. 6, pp. 5171–5183, Jun. 2020.
- [88] G. Nadiradze, A. Sabour, P. Davies, S. Li, and D. Alistarh, "Asynchronous decentralized SGD with quantized and local updates," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 34, 2021, pp. 6829–6842.
- [89] H. Ye, L. Liang, and G. Y. Li, "Decentralized federated learning with unreliable communications," *IEEE J. Sel. Topics Signal Process.*, vol. 16, no. 3, pp. 487–500, Apr. 2022.
- [90] T.-T. Kuo and A. Pham, "Detecting model misconducts in decentralized healthcare federated learning," *Int. J. Med. Informat.*, vol. 158, Feb. 2022, Art. no. 104658.
- [91] P. Vanhaesebrouck, A. Bellet, and M. Tommasi, "Decentralized collaborative learning of personalized models over networks," 2016, *arXiv:1610.05202*.
- [92] A. Gholami, N. Torkzaban, and J. S. Baras, "Trusted decentralized federated learning," in *Proc. IEEE 19th Annu. Consum. Commun. Netw. Conf.*, 2022, pp. 1–6.
- [93] V. Mothukuri, R. M. Parizi, S. Pouriyeh, A. Dehghantanha, and K.-K. R. Choo, "FabricFL: Blockchain-in-the-loop federated learning for trusted decentralized systems," *IEEE Syst. J.*, vol. 16, no. 3, pp. 3711–3722, Sep. 2022.
- [94] J. Konecny et al., "Federated learning: Strategies for improving communication efficiency," 2016, *arXiv:1610.05492*.
- [95] P. M. S. Sánchez, A. H. Celdrán, N. Xie, G. Bovet, G. M. Pérez, and B. Stiller, "FederatedTrust: A solution for trustworthy federated learning," 2023, *arXiv:2302.09844*.
- [96] L. Wang, X. Zhao, Z. Lu, L. Wang, and S. Zhang, "Enhancing privacy preservation and trustworthiness for decentralized federated learning," *Inf. Sci.*, vol. 628, pp. 449–468, May 2023.
- [97] J. Jiang and L. Hu, "Decentralized federated learning with adaptive partial gradient aggregation," *CAAI Trans. Intell. Technol.*, vol. 5, no. 3, pp. 230–236, 2020.
- [98] B. C. Tedeschini et al., "Decentralized federated learning for healthcare networks: A case study on tumor segmentation," *IEEE Access*, vol. 10, pp. 8693–8708, 2022.
- [99] P. Vepakomma, O. Gupta, T. Swedish, and R. Raskar, "Split learning for health: Distributed deep learning without sharing raw patient data," 2018, *arXiv:1812.00564*.
- [100] L. Barbieri, S. Savazzi, and M. Nicoli, "A layer selection optimizer for communication-efficient decentralized federated deep learning," *IEEE Access*, vol. 11, pp. 22155–22173, 2023.
- [101] W. Liu, L. Chen, and W. Zhang, "Decentralized federated learning: Balancing communication and computing costs," *IEEE Trans. Signal Inf. Process. Netw.*, vol. 8, pp. 131–143, 2022.
- [102] W. Liu, L. Chen, Y. Chen, and W. Wang, "Communication-efficient design for quantized decentralized federated learning," 2023, *arXiv:2303.08423*.
- [103] H. Tang, S. Gan, C. Zhang, T. Zhang, and J. Liu, "Communication compression for decentralized training," in *Proc. 32nd Int. Conf. Neural Inf. Process. Syst. (NIPS)*, 2018, pp. 7663–7673.
- [104] K. Yuan, Q. Ling, and W. Yin, "On the convergence of decentralized gradient descent," *SIAM J. Optim.*, vol. 26, no. 3, pp. 1835–1854, 2016.
- [105] W. Shi, Q. Ling, K. Yuan, G. Wu, and W. Yin, "On the linear convergence of the ADMM in decentralized consensus optimization," *IEEE Trans. Signal Process.*, vol. 62, no. 7, pp. 1750–1761, Apr. 2014.
- [106] W. Shi, Q. Ling, G. Wu, and W. Yin, "EXTRA: An exact first-order algorithm for decentralized consensus optimization," *SIAM J. Optim.*, vol. 25, no. 2, pp. 944–966, 2015.
- [107] W. Deng, M.-J. Lai, Z. Peng, and W. Yin, "Parallel multi-block ADMM with $\alpha(1/k)$ convergence," *J. Sci. Comput.*, vol. 71, no. 2, pp. 712–736, 2017.
- [108] S. Wu, D. Huang, and H. Wang, "Network gradient descent algorithm for decentralized federated learning," *J. Bus. Econ. Stat.*, vol. 41, no. 3, pp. 806–818, 2023.
- [109] D. Alistarh, D. Grubic, J. Li, R. Tomioka, and M. Vojnovic, "QSGD: Communication-efficient SGD via gradient quantization and encoding," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, 2017, p. 12.
- [110] D. Basu, D. Data, C. Karakus, and S. Diggavi, *Qsparse-Local-SGD: Distributed SGD With Quantization, Sparsification, and Local Computations*, Curran Assoc. Inc., Red Hook, NY, USA, 2019.
- [111] Y. Shi et al., "Improving the model consistency of decentralized federated learning," 2023, *arXiv:2302.04083*.
- [112] X. Li, Y. Li, J. Wang, C. Chen, L. Yang, and Z. Zheng, "Decentralized federated meta-learning framework for few-shot multitask learning," *Int. J. Intell. Syst.*, vol. 37, no. 11, pp. 8490–8522, 2022.
- [113] R. Anil, G. Pereyra, A. T. Passos, R. Ormandi, G. Dahl, and G. Hinton, "Large scale distributed neural network training through online distillation," in *Proc. 6th Int. Conf. Learn. Rep. (ICLR)*, 2018, p. 9.
- [114] H. Seo, J. Park, S. Oh, M. Bennis, and S.-L. Kim, "Federated knowledge distillation," 2020, *arXiv:2011.02367*.
- [115] C. Wu, F. Wu, L. Lyu, Y. Huang, and X. Xie, "Communication-efficient federated learning via knowledge distillation," *Nat. Commun.*, vol. 13, no. 1, p. 2032, 2022.
- [116] I. Shiri et al., "Decentralized distributed multi-institutional pet image segmentation using a federated deep learning framework," *Clin. Nucl. Med.*, vol. 47, no. 7, pp. 606–617, 2022.
- [117] K. Hsieh, A. Phanishayee, O. Mutlu, and P. B. Gibbons, "The non-IID data quagmire of decentralized machine learning," in *Proc. 37th Int. Conf. Mach. Learn. (ICML)*, 2020, pp. 4387–4398.
- [118] H. Gao, M. T. Thai, and J. Wu, "When decentralized optimization meets federated learning," *IEEE Netw.*, early access, Feb. 7, 2023, doi: [10.1109/MNET.132.2200530](https://doi.org/10.1109/MNET.132.2200530).
- [119] Y. Shi et al., "Towards more suitable personalization in federated learning via decentralized partial model training," 2023, *arXiv:2305.15157*.
- [120] M. Duan et al., "FedGroup: Efficient federated learning via decomposed similarity-based clustering," in *Proc. IEEE Int. Conf. Parallel Distrib. Process. Appl. Big Data Cloud Comput. Sustain. Comput. Commun. Soc. Comput. Netw.*, 2021, pp. 228–237.
- [121] C. Hu, J. Jiang, and Z. Wang, "Decentralized federated learning: A segmented gossip approach," 2019, *arXiv:1908.07782*.
- [122] A. Schwab and J. Lunze, "A distributed algorithm to maintain a proximity communication network among mobile agents using the delaunay triangulation," *Eur. J. Control.*, vol. 60, pp. 125–134, Jul. 2021.
- [123] Y. Ye, H. Chen, Z. Ma, and M. Xiao, "Decentralized consensus optimization based on parallel random walk," *IEEE Commun. Lett.*, vol. 24, no. 2, pp. 391–395, Feb. 2020.
- [124] N. R. Barroso, D. Jiménez-López, M. V. Luzón, F. Herrera, and E. Martínez-Cámara, "Survey on federated learning threats: Concepts, taxonomy on attacks and defences, experimental study and challenges," *Inf. Fusion*, vol. 90, pp. 148–173, Feb. 2023.
- [125] J. Wang, X. Chang, R. J. Rodriguez, and Y. Wang, "Assessing anonymous and selfish free-rider attacks in federated learning," in *Proc. IEEE Symp. Comput. Commun.*, 2022, pp. 1–6.
- [126] H. Wang, L. M. González, D. Eklund, and S. Raza, "Non-IID data rebalancing at IoT edge with peer-to-peer federated learning for anomaly detection," in *Proc. 14th ACM Conf. Security Privacy Wireless Mobile Netw. (WiSec)*, 2021, pp. 153–163.
- [127] X. Ma and D. Xu, "Torr: A lightweight blockchain for decentralized federated learning," *IEEE Internet Things J.*, early access, Jun. 1, 2023, doi: [10.1109/IJOT.2023.3288078](https://doi.org/10.1109/IJOT.2023.3288078).
- [128] L. Huang, A. L. Shea, H. Qian, A. Masurkar, H. Deng, and D. Liu, "Patient clustering improves efficiency of federated machine learning to predict mortality and hospital stay time using distributed electronic medical records," *J. Biomed. Informat.*, vol. 99, Nov. 2019, Art. no. 103291.
- [129] A. Nedic and A. Ozdaglar, "Distributed subgradient methods for multi-agent optimization," *IEEE Trans. Autom. Control*, vol. 54, no. 1, pp. 48–61, Jan. 2009.
- [130] Z. Chen, D. Li, J. Zhu, and S. Zhang, "DACFL: Dynamic average consensus-based federated learning in decentralized sensors network," *Sensors*, vol. 22, no. 9, p. 3317, 2022.
- [131] T. Sun, D. Li, and B. Wang, "Decentralized federated averaging," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 4, pp. 4289–4301, Apr. 2023.
- [132] Y. Yuan et al., "DeceFL: A principled fully decentralized federated learning framework," *Nat. Sci. Open*, vol. 2, no. 1, 2023, Art. no. 20220043.
- [133] S. Minsker and N. Strawn, "Distributed statistical estimation and rates of convergence in normal approximation," 2017, *arXiv:1704.02658*.
- [134] X. Mao, K. Yuan, Y. Hu, Y. Gu, A. H. Sayed, and W. Yin, "Walkman: A communication-efficient random-walk algorithm for decentralized optimization," *IEEE Trans. Signal Process.*, vol. 68, pp. 2513–2528, 2020.

- [135] X. Mao, Y. Gu, and W. Yin, "Walk proximal gradient: An energy-efficient algorithm for consensus optimization," *IEEE Internet Things J.*, vol. 6, no. 2, pp. 2048–2060, Apr. 2019.
- [136] D. C. Nguyen, S. Hosseinalipour, D. J. Love, P. N. Pathirana, and C. G. Brinton, "Latency optimization for blockchain-empowered federated learning in multi-server edge computing," *IEEE J. Sel. Areas Commun.*, vol. 40, no. 12, pp. 3373–3390, Dec. 2022.
- [137] S. Zhu and B. Chen, "Quantized consensus by the ADMM: Probabilistic versus deterministic quantizers," *IEEE Trans. Signal Process.*, vol. 64, no. 7, pp. 1700–1713, Apr. 2016.
- [138] S. K. Singh, L. T. Yang, and J. H. Park, "FusionFedBlock: Fusion of blockchain and federated learning to preserve privacy in industry 5.0," *Inf. Fusion*, vol. 90, pp. 233–240, Feb. 2023.
- [139] I. A. Ridhawi, S. Otoum, and M. Aloqaily, "Decentralized zero-trust framework for digital twin-based 6G," 2023, *arXiv:2302.03107*.
- [140] X. Lian, C. Zhang, H. Zhang, C.-J. Hsieh, W. Zhang, and J. Liu, "Can decentralized algorithms outperform centralized algorithms? A case study for decentralized parallel stochastic gradient descent," in *Proc. 31st Int. Conf. Neural Inf. Process. Syst. (NIPS)*, 2017, pp. 5336–5346.
- [141] G. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," 2015, *arXiv:1503.02531*.
- [142] M. H. Rehman, A. M. Dirir, K. Salah, E. Damiani, and D. Svetinovic, "TrustFed: A framework for fair and trustworthy cross-device federated learning in IIoT," *IEEE Trans. Ind. Informat.*, vol. 17, no. 12, pp. 8485–8494, Dec. 2021.
- [143] Google. "Tensorflow federated." 2022. [Online]. Available: <https://www.tensorflow.org/federated>
- [144] A. Ziller et al., "PySyft: A library for easy federated learning," in *Federated Learning Systems: Towards Next-Generation AI*, M. H. u. Rehman and M. M. Gaber, Eds. Cham, Switzerland: Springer Int., 2021, pp. 111–139, doi: [10.1007/978-3-030-70604-3_5](https://doi.org/10.1007/978-3-030-70604-3_5).
- [145] Y. Xie et al., "FederatedScope: A flexible federated learning platform for heterogeneity," 2022, *arXiv:2204.05011*.
- [146] C. He et al., "FedML: A research library and benchmark for federated machine learning," in *Proc. Adv. Neural Inf. Process. Syst.*, 2020, p. 9.
- [147] S. Caldas et al., "LEAF: A benchmark for federated settings," 2018, *arXiv:1812.01097*.
- [148] A. G. Roy, S. Siddiqui, S. Pölsterl, N. Navab, and C. Wachinger, "BrainTorrent: A peer-to-peer environment for decentralized federated learning," 2019, *arXiv:1905.06731*.
- [149] C. Pappas, D. Chatzopoulos, S. Lalis, and M. Vavalis, "IPLS: A framework for decentralized federated learning," in *Proc. IFIP Netw. Conf.*, 2021, pp. 1–6.
- [150] M. Ghanem, F. Dawoud, H. Gamal, E. Soliman, H. Sharara, and T. El-Batt, "FLOBC: A decentralized blockchain-based federated learning framework," 2021, *arXiv:2112.11873*.
- [151] J. Li et al., "Blockchain assisted decentralized federated learning (BLADE-FL): Performance analysis and resource allocation," *IEEE Trans. Parallel Distrib. Syst.*, vol. 33, no. 10, pp. 2401–2415, May 2022.
- [152] C. Che, X. Li, C. Chen, X. He, and Z. Zheng, "A decentralized federated learning framework via committee mechanism with convergence guarantee," *IEEE Trans. Parallel Distrib. Syst.*, vol. 33, no. 12, pp. 4783–4800, Dec. 2022.
- [153] E. T. M. Beltrán et al., "Fedstellar: A platform for decentralized federated learning," 2023, *arXiv:2306.09750*.
- [154] A. Krizhevsky, "Learning multiple layers of features from tiny images," 2009. [Online]. Available: <https://www.cs.toronto.edu/~kriz/learning-features-2009-TR.pdf>
- [155] A. Go, R. Bhayani, and L. Huang, "Twitter sentiment classification using distant supervision," 2009. [Online]. Available: <https://www-cs.stanford.edu/people/alecmgo/papers/TwitterDistantSupervision09.pdf>
- [156] D. Gurung, S. R. Pokhrel, and G. Li, "Decentralized quantum federated learning for metaverse: Analysis, design and implementation," 2023, *arXiv:2306.11297*.
- [157] X. Zhou et al., "Decentralized P2P federated learning for privacy-preserving and resilient mobile robotic systems," *IEEE Wireless Commun.*, vol. 30, no. 2, pp. 82–89, Apr. 2023.
- [158] Q. Liu et al., "Asynchronous decentralized federated learning for collaborative fault diagnosis of PV stations," *IEEE Trans. Netw. Sci. Eng.*, vol. 9, no. 3, pp. 1680–1696, May/Jun. 2022.
- [159] M. Nakip, B. C. Gül, and E. Gelenbe, "Decentralized online federated g-network learning for lightweight intrusion detection," 2023, *arXiv:2306.13029*, 2023.
- [160] D. Liu, T. Miller, R. Sayeed, and K. D. Mandl, "FADL: Federated-autonomous deep learning for distributed electronic health record," 2018. [Online]. Available: <https://arxiv.org/pdf/1811.11400.pdf>
- [161] T. J. Pollard, A. E. W. Johnson, J. D. Raffa, L. A. Celi, R. G. Mark, and O. Badawi, "The eICU collaborative research database, a freely available multi-center database for critical care research," *Sci. Data*, vol. 5, no. 1, 2018, Art. no. 180178.
- [162] S. Lu, Y. Zhang, and Y. Wang, "Decentralized federated learning for electronic health records," in *Proc. 54th Annu. Conf. Inf. Sci. Syst.*, 2020, pp. 1–5.
- [163] H. Elayan, M. Aloqaily, and M. Guizani, "Deep federated learning for IoT-based decentralized healthcare systems," in *Proc. Int. Wireless Commun. Mobile Comput.*, 2021, pp. 105–109.
- [164] S. Wamat-Herresthal et al., "Swarm learning for decentralized and confidential clinical machine learning," *Nature*, vol. 594, no. 7862, pp. 265–270, 2021.
- [165] "Geo dataset browser." 2016. [Online]. Available: <https://www.ncbi.nlm.nih.gov/sites/GDSbrowser/>
- [166] X. Wang, Y. Peng, L. Lu, Z. Lu, M. Bagheri, and R. M. Summers, "ChestX-Ray8: Hospital-scale chest X-ray database and benchmarks on weakly-supervised classification and localization of common thorax diseases," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2017, pp. 3462–3471, doi: [10.1109/CVPR.2017.369](https://doi.org/10.1109/CVPR.2017.369).
- [167] M. M. Salim and J. H. Park, "Federated learning-based secure electronic health record sharing scheme in medical informatics," *IEEE J. Biomed. Health Informat.*, vol. 27, no. 2, pp. 617–624, Feb. 2023.
- [168] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2009, pp. 248–255.
- [169] B. H. Menze et al., "The multimodal brain tumor image segmentation benchmark (brats)," *IEEE Trans. Med. Imag.*, vol. 34, no. 10, pp. 1993–2024, Oct. 2015.
- [170] K. H. Zou, A. Liu, A. I. Bandos, L. Ohno-Machado, and H. E. Rockette, *Statistical Evaluation of Diagnostic Performance*, 1st ed. Boca Raton, FL, USA: CRC, 2011.
- [171] X. Yang, X. He, J. Zhao, Y. Zhang, S. Zhang, and P. Xie, "COVID-CT-dataset: A CT scan dataset about COVID-19," 2020, *arXiv:2003.13865*.
- [172] A. Dos. "Predicting human eye diseases." 2021. [Online]. Available: <https://www.kaggle.com/datasets/fabianogalaxy/dataset-with-cataracts-images>
- [173] M. Jennings, "Skin-cancer-identification." 2021. [Online]. Available: <https://github.com/Matt-Jennings-GitHub/Skin-Cancer-Identification>
- [174] L. Deng, "The MNIST database of handwritten digit images for machine learning research," *IEEE Signal Process. Mag.*, vol. 29, no. 6, pp. 141–142, Nov. 2012.
- [175] M. Pennisi et al., "Decentralized distributed learning with privacy-preserving data synthesis," 2022, *arXiv:2206.10048*.
- [176] S. Candemir et al., "Lung segmentation in chest radiographs using anatomical atlases with nonrigid registration," *IEEE Trans. Med. Imag.*, vol. 33, no. 2, pp. 577–590, Feb. 2014.
- [177] S. Jaeger et al., "Automatic tuberculosis screening using chest radiographs," *IEEE Trans. Med. Imag.*, vol. 33, no. 2, pp. 233–245, Feb. 2014.
- [178] P. Tschandl, C. Rosendahl, and H. Kittler, "The HAM10000 dataset, a large collection of multi-source dermatoscopic images of common pigmented skin lesions," *Sci. Data*, vol. 5, no. 1, 2018, Art. no. 180161.
- [179] Y. Tian, S. Wang, J. Xiong, R. Bi, Z. Zhou, and M. Z. A. Bhuiyan, "Robust and privacy-preserving decentralized deep federated learning training: Focusing on digital healthcare applications," *IEEE/ACM Trans. Comput. Biol. Bioinform.*, early access, Mar. 3, 2023, doi: [10.1109/TCBB.2023.3243932](https://doi.org/10.1109/TCBB.2023.3243932).
- [180] S. Savazzi. "Federated learning: mmWave MIMO radar dataset for testing." 2020. [Online]. Available: <https://iee-dataport.org/open-access/federated-learning-mmwave-mimo-radar-dataset-testing>
- [181] Y. Qu, S. R. Pokhrel, S. Garg, L. Gao, and Y. Xiang, "A blockchain federated learning framework for cognitive computing in industry 4.0 networks," *IEEE Trans. Ind. Informat.*, vol. 17, no. 4, pp. 2964–2973, Apr. 2021.
- [182] Y. Meidan, M. Bohadana, Y. Mathov, Y. Mirsky, A. Shabtai, D. Breitenbacher, and Y. Elovici, "N-BaloT: Network-based detection of IoT botnet attacks using deep autoencoders," *IEEE Pervasive Comput.*, vol. 17, no. 3, pp. 12–22, Jul.–Sep. 2018.
- [183] Z. Lian and C. Su, "Decentralized federated learning for Internet of Things anomaly detection," in *Proc. ACM Asia Conf. Comput. Commun. Security*, 2022, pp. 1249–1251.
- [184] S. Garcia, A. Parmisano, and M. J. Erquiaga, "IoT-23: A labeled dataset with malicious and benign IoT network traffic." 2020. [Online]. Available: <https://paperswithcode.com/dataset/iot-23>

- [185] M. Abdel-Basset, N. Moustafa, and H. Hawash, "Privacy-preserved cyberattack detection in Industrial Edge of Things (IEoT): A blockchain-orchestrated federated learning approach," *IEEE Trans. Ind. Informat.*, vol. 18, no. 11, pp. 7920–7934, Nov. 2022.
- [186] A. Alsaedi, N. Moustafa, Z. Tari, A. Mahmood, and A. Anwar, "TON_IoT telemetry dataset: A new generation dataset of IoT and IIoT for data-driven intrusion detection systems," *IEEE Access*, vol. 8, pp. 165130–165150, 2020.
- [187] R. Damasevicius et al., "LITNET-2020: An annotated real-world network flow dataset for network intrusion detection," *Electronics*, vol. 9, no. 5, p. 800, 2020.
- [188] H. Ochiai, Y. Sun, Q. Jin, N. Wongwiwatchai, and H. Esaki, "Wireless ad hoc federated learning: A fully distributed cooperative machine learning," 2022, *arXiv:2205.11779*.
- [189] T. Ranathunga, A. McGibney, S. Rea, and S. Bharti, "Blockchain-based decentralized model aggregation for cross-silo federated learning in industry 4.0," *IEEE Internet Things J.*, vol. 10, no. 5, pp. 4449–4461, Mar. 2023.
- [190] D. Monschein, J. A. P. Pérez, T. Piotrowski, Z. Nocht, O. P. Waldhorst, and C. Zarpins, "Towards a peer-to-peer federated machine learning environment for continuous authentication," in *Proc. IEEE Symp. Comput. Commun.*, 2021, pp. 1–6.
- [191] Y. Lu, X. Huang, K. Zhang, S. Maharjan, and Y. Zhang, "Communication-efficient federated learning and permissioned blockchain for digital twin edge networks," *IEEE Internet Things J.*, vol. 8, no. 4, pp. 2276–2288, Feb. 2021.
- [192] H. Xiao, K. Rasul, and R. Vollgraf, "Fashion-MNIST: A novel image dataset for benchmarking machine learning algorithms," 2017, *arXiv:1708.07747*.
- [193] F. Wilhelm, E. Guerra, and P. Dini, "On the decentralization of blockchain-enabled asynchronous federated learning," 2022, *arXiv:2205.10201*.
- [194] G. Cohen, S. Afshar, J. Tapson, and A. van Schaik, "EMNIST: An extension of MNIST to handwritten letters," 2017, *arXiv:1702.05373*.
- [195] D. Yang, D. Zhang, V. W. Zheng, and Z. Yu, "Modeling user activity preference by leveraging user spatial temporal characteristics in LBSNs," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 45, no. 1, pp. 129–142, Jan. 2015.
- [196] E. Cho, S. A. Myers, and J. Leskovec, "Friendship and mobility: User movement in location-based social networks," in *Proc. 17th ACM SIGKDD Int. Conf. Knowl. Disc. Data Min.*, 2011, pp. 1082–1090.
- [197] F. M. Harper and J. A. Konstan, "The MovieLens datasets: History and context," *ACM Trans. Interact. Intell. Syst.*, vol. 5, no. 4, pp. 1–19, 2015.
- [198] B. Rozemberczki, C. Allen, and R. Sarkar, "Multi-Scale attributed node embedding," *J. Complex Netw.*, vol. 9, no. 2, 2021, Art. no. cnab014.
- [199] Q. Pan, Y. Zhu, and L. Chu, "LUMOS: Heterogeneity-aware federated graph learning over decentralized devices," 2023, *arXiv:2303.00492*.
- [200] B. Rozemberczki and R. Sarkar, "Characteristic functions on graphs: Birds of a feather, from statistical descriptors to parametric models," in *Proc. ACM CIKM*, 2020, pp. 1325–1334.
- [201] I. Arapakis, P. Papadopoulos, K. Katevas, and D. Perino, "P4I: Privacy preserving peer-to-peer learning for infrastructureless setups," 2023, *arXiv:2302.13438*.
- [202] "Avito context ad clicks." 2016. [Online]. Available: <https://www.kaggle.com/c/avito-context-ad-clicks>
- [203] A. L. Maas, R. E. Daly, P. T. Pham, D. Huang, A. Y. Ng, and C. Potts, "Learning word vectors for sentiment analysis," in *Proc. 49th Annu. Meeting Assoc. Comput. Linguist. Human Lang. Technol.*, 2011, pp. 142–150.
- [204] E. Jeong and M. Kountouris, "Personalized decentralized federated learning with knowledge distillation," 2023, *arXiv:2302.12156*.
- [205] "IoT device identification." 2021. [Online]. Available: <https://www.kaggle.com/datasets/fanbyprinciple/iot-device-identification>
- [206] A. Salama, A. Stergioulis, A. M. Hayajneh, S. A. R. Zaidi, D. McLernon, and I. Robertson, "Decentralized federated learning over slotted aloha wireless mesh networking," *IEEE Access*, vol. 11, pp. 18326–18342, 2023.
- [207] W. Koch, "On digital ethics for artificial intelligence in hybrid military operations," 2021. [Online]. Available: <https://www.sto.nato.int/publications/S%20Meeti%20Proceedings/STO-MP-IST-190/MP-IS-T-190-22.pdf>
- [208] N. I. Mowla, N. H. Tran, I. Doh, and K. Chae, "Federated learning-based cognitive detection of jamming attack in flying ad-hoc network," *IEEE Access*, vol. 8, pp. 4338–4350, 2020.
- [209] O. Punal, C. Pereira, A. Aguiar, and J. Gross, "CRAWDAD dataset uportorwthaachen/vanetjamming2014 (v.2014-05-12)," 2014. [Online]. Available: <https://iee-dataport.org/collections/crawdadd?page=15>
- [210] G. F. Riley and T. R. Henderson, "The ns-3 network simulator," in *Modeling and Tools for Network Simulation*, K. Wehrle, M. Güneş, and J. Gross, Eds. Berlin, Germany: Springer, 2010, pp. 15–34, doi: [10.1007/978-3-642-12331-3_2](https://doi.org/10.1007/978-3-642-12331-3_2).
- [211] P. K. Sharma, J. H. Park, and K. Cho, "Blockchain and federated learning-based distributed computing defence framework for sustainable society," *Sustain. Cities Soc.*, vol. 59, Aug. 2020, Art. no. 102220.
- [212] Y. Qu et al., "Decentralized federated learning for UAV networks: Architecture, challenges, and opportunities," *IEEE Netw.*, vol. 35, no. 6, pp. 156–162, Nov./Dec. 2021.
- [213] C. Zhu, X. Zhu, J. Ren, and T. Qin, "Blockchain-enabled federated learning for UAV edge computing network: Issues and solutions," *IEEE Access*, vol. 10, pp. 56591–56610, 2022.
- [214] M. S. Al-Abiad, M. Hassan, and M. Hossain, "Decentralized model dissemination empowered federated learning in mmWave aerial-terrestrial integrated networks," 2023, *arXiv:2303.00032*.
- [215] A. Giannopoulos, P. Gkonis, P. Bithas, N. Nomikos, G. Ntroulias, and P. Trakadas, "Federated learning for maritime environments: use cases, experimental results, and open issues," 2023, *arXiv:2213.3549.v1*.
- [216] C. Reiser, "Creiser/drone-detection." 2022. [Online]. Available: <https://github.com/creiser/drone-detection>
- [217] R. Fisher, "CVONLINE: Image databases." 2019. [Online]. Available: <https://homepages.inf.ed.ac.uk/rbf/CVonline/Imagedbase.htm>
- [218] E. T. M. Beltrín, M. Q. Pérez, S. L. Bernal, G. M. Pérez, and A. H. Celdrán, "SAFE CAR: A brain-computer interface and intelligent framework to detect drivers distractions," *Exp. Syst. Appl.*, vol. 203, Mar. 2022, Art. no. 117402.
- [219] Y. Lu, X. Huang, K. Zhang, S. Maharjan, and Y. Zhang, "Blockchain empowered asynchronous federated learning for secure data sharing in Internet of Vehicles," *IEEE Trans. Veh. Technol.*, vol. 69, no. 4, pp. 4298–4311, Apr. 2020.
- [220] S. R. Pokhrel and J. Choi, "Federated learning with blockchain for autonomous vehicles: Analysis and design challenges," *IEEE Trans. Commun.*, vol. 68, no. 8, pp. 4734–4746, Aug. 2020.
- [221] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? The Kitti vision benchmark suite," in *Proc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2012, pp. 3354–3361.
- [222] H. Caesar et al., "nuScenes: A multimodal dataset for autonomous driving," 2019, *arXiv:1903.11027*.
- [223] F. Bragato et al., "Towards decentralized predictive quality of service in next-generation vehicular networks," 2023, *arXiv:2302.11268*.
- [224] X. Hu, R. Li, Y. Ning, K. Ota, and L. Wang, "A data sharing scheme based on federated learning in IoV," *IEEE Trans. Veh. Technol.*, vol. 72, no. 9, pp. 11644–11656, Sep. 2023.

Flighter: Decentralized Federated Learning and Situational Awareness for Secure Military Aerial Reconnaissance



Title:	Flighter: Decentralized Federated Learning and Situational Awareness for Secure Military Aerial Reconnaissance
Authors:	Enrique Tomás Martínez Beltrán, Pedro Miguel Sánchez Sánchez, G�r�me Bovet, Burkhard Stiller, Gregorio Mart�nez P�rez, Alberto Huertas Celdr�n.
Journal:	IEEE Communications Magazine
JIF:	8.3 Q1-D1 (2024)
Publisher:	IEEE
Volume:	63
Number:	10
Pages:	136–142
Month:	Oct
Year:	2025
DOI:	10.1109/MCOM.003.2400401
Status:	Published

Abstract

Mosaic warfare is a military strategy where reconnaissance missions with aerial vehicles are critical for gathering enemy information and achieving battlefield dominance. Nowadays, machine learning (ML) techniques play a pivotal role in this task by enabling precise detection of military vehicles. However, reconnaissance missions face challenges, particularly when enemies launch attacks targeting ML models and aerial exploration vehicles. Combining decentralized federated learning (DFL) and situational awareness could improve these challenges by facilitating collaborative, context-aware, robust, and privacy-preserving ML-based learning processes. Nevertheless, existing literature overlooks these research fields and their applicability to military reconnaissance missions. Thus, this work introduces Flighter, a novel framework that trains DFL models for enemy detection while safeguarding model integrity using a situational awareness-based defense module. Flighter considers model similarity, flight formations, geopositioning, resource usage, and interconnection rate between aerial vehicles to defend against adversarial attacks targeting reconnaissance tasks. A simulated military scenario using synthetic aperture radar (SAR) data of military vehicles has been deployed to evaluate the framework performance. The evaluation involved

four military aircraft operating in two flight formations under adversarial attacks affecting aircraft geopositioning and collaborative training. Extensive experimentation with different attacks demonstrated that Flighter enhances literature detection accuracy and time without compromising privacy.

Keywords

Decentralized Federated Learning · Situational Awareness · Military Aerial Reconnaissance
· Adversarial Attacks · Synthetic Aperture Radar

MILITARY COMMUNICATIONS & NETWORKS

Flighter: Decentralized Federated Learning and Situational Awareness for Secure Military Aerial Reconnaissance

Enrique Tomás Martínez Beltrán, Pedro Miguel Sánchez Sánchez, G  r  me Bovet, Burkhard Stiller, Gregorio Mart  nez P  rez, and Alberto Huertas Celdr  n

ABSTRACT

Mosaic warfare is a military strategy where reconnaissance missions with aerial vehicles are critical for gathering enemy information and achieving battlefield dominance. Nowadays, machine learning (ML) techniques play a pivotal role in this task by enabling precise detection of military vehicles. However, reconnaissance missions face challenges, particularly when enemies launch attacks targeting ML models and aerial exploration vehicles. Combining decentralized federated learning (DFL) and situational awareness could improve these challenges by facilitating collaborative, context-aware, robust, and privacy-preserving ML-based learning processes. Nevertheless, existing literature overlooks these research fields and their applicability to military reconnaissance missions. Thus, this work introduces Flighter, a novel framework that trains DFL models for enemy detection while safeguarding model integrity using a situational awareness-based defense module. Flighter considers model similarity, flight formations, geopositioning, resource usage, and interconnection rate between aerial vehicles to defend against adversarial attacks targeting reconnaissance tasks. A simulated military scenario using synthetic aperture radar (SAR) data of military vehicles has been deployed to evaluate the framework performance. The evaluation involved four military aircraft operating in two flight formations under adversarial attacks affecting aircraft geopositioning and collaborative training. Extensive experimentation with different attacks demonstrated that Flighter enhances literature detection accuracy and time without compromising privacy.

INTRODUCTION

Modern military operations, characterized by strategies like mosaic warfare, require rapid, adaptive, and precise reconnaissance capabilities to maintain battlefield dominance [1]. These military missions, including aerial reconnaissance to detect enemy vehicles or monitor enemy activities, demand intelligence, robustness, and adaptability. The goal of detection is to confirm the exact nature of the observed target, distinguishing it from other possible targets or entities. In battlefield scenarios,

integrating ML techniques into detection systems becomes increasingly significant for enhancing accurate, fast, and automatic detection in reconnaissance operations [2]. These systems must accurately identify targets while ensuring data privacy and security to safeguard sensitive information. Also, they must remain adaptable to changing environments, where conditions can shift unpredictably. Finally, resilience against adversarial attacks, which aim to compromise model integrity or disrupt reconnaissance operations, is critical for mission success in hostile environments.

Federated Learning (FL), proposed by Google in 2016, deals with some of these requirements by training collaborative and privacy-preserving ML models. In FL, each participant trains a local model using its data and then sends only the model updates (such as weights or gradients) to a central server. These updates are then aggregated to form a global model, which is shared back with all participants [3]. This process is repeated for several rounds until the model converges. Within the military reconnaissance field, FL has been used in Unmanned Aerial Vehicles (UAVs) to collaboratively learn and improve enemy detection algorithms while keeping local data confidential [4]. However, FL-based solutions used in traditional military environments rely on a central device to coordinate the learning process, which poses significant drawbacks, such as a single point of failure and synchronization bottlenecks during model updates [5]. In aerial reconnaissance missions, a central server is often impractical due to the nature of the battlefield, where secure and reliable communication links are frequently disrupted or non-existent. Additionally, the rapid movement of aerial vehicles makes it difficult to maintain consistent connectivity to a central server.

DFL [3] improves the previous limitations by allowing each participant to train local models and contribute to the global model through decentralized aggregation. Participants share updates directly with each other, ensuring continuous collaboration in the absence of a server. Decentralization is particularly advantageous for military aerial reconnaissance missions operating in highly dynamic environments characterized by complex

Enrique Tom  s Mart  nez Beltr  n (corresponding author), Pedro Miguel S  nchez S  nchez, Gregorio Mart  nez P  rez, and Alberto Huertas Celdr  n are with University of Murcia, Spain; Alberto Huertas Celdr  n and Burkhard Stiller are with the University of Zurich UZH, Switzerland; G  r  me Bovet is with the Cyber-Defence Campus within amasuisse Science & Technology, Switzerland.

Digital Object Identifier: 10.1109/MCOM.003.2400401

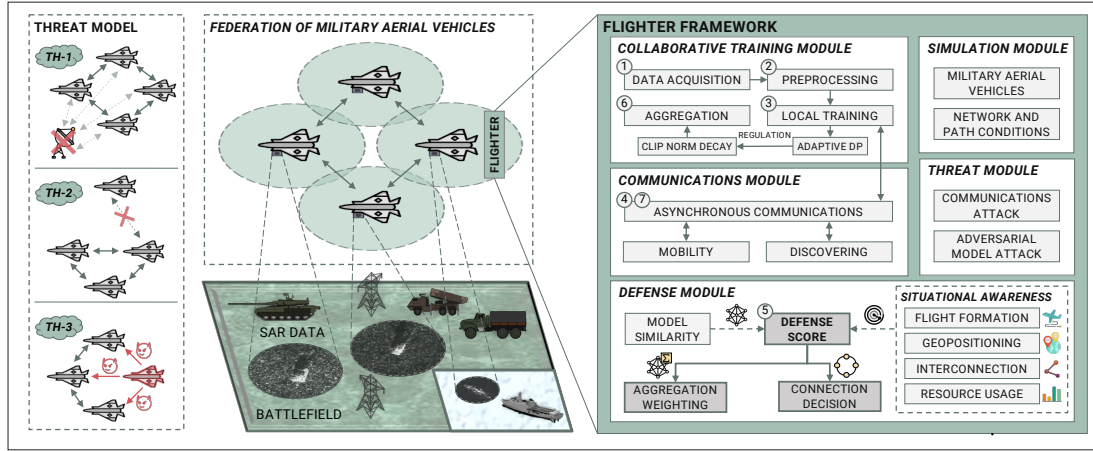


FIGURE 1. Overview of the Flighter framework.

maneuvers, and the ever-present risk of adversarial interference [6]. To maximize the effectiveness of DFL in such missions, robust defense mechanisms are essential for ensuring reliable and secure information exchange. Since DFL involves direct model sharing between participants, privacy-preserving techniques are crucial to protect sensitive data and minimize the risk of privacy leakage during the exchanges. These mechanisms are implemented by verifying model update consistency, communication behavior, and adherence to designated flight paths [7]. By integrating situational awareness into DFL, previous defense mechanisms are further strengthened, enabling real-time, informed decision-making based on a comprehensive understanding of the battlefield [8]. Then, military vehicles can accurately perceive, comprehend, and project their environment, which is crucial for effective enemy detection and overall mission success. In military aerial missions, situational awareness involves key capabilities such as flight formation, geopositioning, resource usage, and average vehicle interconnections [2]. Each capability ensures that aircrafts maintain designated positions, optimize operational resources, and reliably communicate within the squadron, thereby enhancing the robustness and security of the federation.

Despite the benefits of integrating defense mechanisms and situational awareness with DFL, existing literature has only partially addressed these challenges. Some studies have proposed techniques such as blockchain for decentralization and secure transaction records [9], as well as implementing advanced aggregation algorithms to mitigate malicious contributions [8]. However, previous approaches often have limitations, such as relying on static network topologies or lacking decentralization in key functions like model update aggregation [4]. Additionally, they do not fully account for the highly dynamic nature of military air missions, leading to communication disruptions and inconsistent updates in DFL [8]. Many existing solutions also focus narrowly on specific attack types without adopting a comprehensive, adaptive security strategy [6]. Although some research has explored the applicability of FL in military contexts, there remain significant chal-

lenges to situational awareness, model convergence, and adversarial threats. To improve these challenges, this work presents:

- A comprehensive threat model tailored to the challenges of DFL in dynamic scenarios. This model considers the aerial missions, addressing communication disruptions and adversarial cyberattacks that could impact DFL operations and the timely detection of enemy vehicles.
- Flighter, a robust DFL framework for military aerial reconnaissance missions [10]. It integrates a situational-awareness-based defense module to ensure secure and decentralized training of federated models, effectively addressing the unique challenges of dynamic military environments. Moreover, it enhances model resilience by mitigating attacks while maintaining performance in adverse conditions.
- A simulated military scenario with four aircraft equipped with SAR in different flight formations to detect enemy vehicles. Extensive simulations, including geopositioning deviation, collision course manipulation, and adversarial data poisoning attacks, showed effective mitigation of communication attacks and enhanced model robustness, achieving F1 score up to 95.8%; and maintained efficient resource usage and response times, with CPU around 57%, RAM at 1.7 GB, and reducing packet loss while maintaining stable round times.

THREATS IN FL-BASED MILITARY RECONNAISSANCE MISSIONS

This section presents the threats encountered in DFL-assisted reconnaissance missions, considered the most critical due to their potential impact on mission success, as illustrated on the left side of Fig. 1.

TH-1: COMMAND AND CONTROL WEAKNESSES

Military aerial vehicles rely on decentralized communication for managing and synchronizing model updates instead of a central command server. This decentralization can cause inconsistencies in model performance and synchronization challenges during missions [5]. The absence of a central server makes it challenging to detect

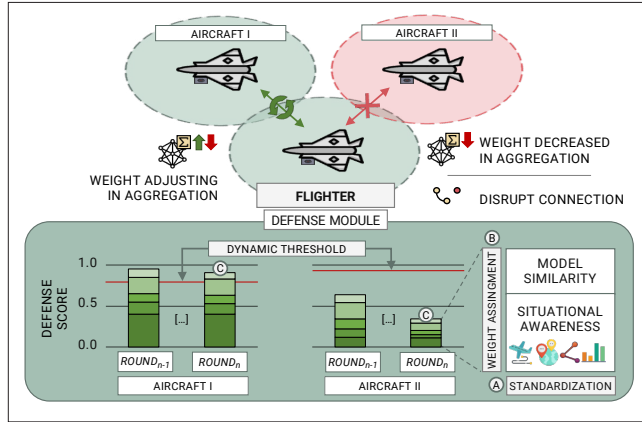


FIGURE 2. Overview of the defense module in Flighter.

and mitigate these issues promptly, allowing a single rogue vehicle to compromise the entire operational effectiveness. A high-stakes military scenario could result in failed reconnaissance missions and decreased intelligence.

TH-2: OPERATIONAL MOBILITY HAZARDS

Aircraft in formation constantly change positions, complicating the maintenance of secure and reliable communications [4]. Adversaries could exploit the movements to interfere or hijack communication channels. For instance, an adversary manipulating an aircraft trajectory through cyberattacks on its navigation systems could force it to deviate from its intended path [8]. Such deviations could lead to physical collisions, isolation of the aircraft from the network, or compromise the federation across the entire squadron.

TH-3: ROGUE VEHICLE SABOTAGE

Aerial vehicles within the network may be compromised or act maliciously, launching adversarial attacks such as data poisoning, where false data is injected to degrade the model performance. Malicious aerial vehicles can also craft inputs to mislead the learning algorithm, resulting in incorrect predictions. These activities significantly impair system effectiveness, compromising mission success by causing the model to overlook enemy positions or misidentify friendly units as hostile, directly impacting military operations [7].

FLIGHTER FRAMEWORK

Flighter improves the limitations of existing FL solutions for military aerial reconnaissance missions by incorporating DFL and an advanced defense module based on situational awareness [10]. It effectively counters TH-1, TH-2, and TH-3. The framework leverages decentralization to enhance resilience and operational effectiveness, allowing aerial vehicles to operate independently while collaboratively and securely training models. Flighter comprises the following modules (Fig. 1): collaborative training, communications, and defense modules, along with two additional modules for scenarios simulation and threat generation.

The core of Flighter consists of collaborative training, communications, and defense mod-

ules, which manage local model training, data exchange, and system protection, respectively. Initially, each aerial vehicle collects and preprocesses data from its onboard sensors, such as SAR data for target detection ①, and engages in local model training ②, ③. Once local models are trained, aerial vehicles periodically share their model updates with neighbors ④. To protect sensitive data, Flighter integrates an adaptive Differential Privacy (DP), which selectively applies privacy constraints by retaining critical parameters and adding noise to less significant ones [11]. This improves privacy without compromising model performance or convergence. DP combines differential constraint strategies and a clip norm decay to defend against privacy leakage, poisoning, and backdoor attacks. The clip norm decay reduces injected noise by gradually decreasing the clipping threshold of model updates throughout training, preserving utility while enhancing protection. Moreover, Flighter incorporates a sophisticated communications module that utilizes a fault-tolerant asynchronous communication mechanism, supporting various network topologies and mobile participants. It leverages chunk-based data transmission adapted to network conditions with traceability for packet loss. The communication module uses LZ4 compression, proving an optimal balance between speed and efficiency [12]. Discovery messages dynamically re-establish connections as aircraft move, enhancing mobility management. If disconnections occur due to slow response times or degraded performance, Flighter proactively initiates reconnections to maintain seamless network integration. Additionally, a congestion mechanism reroutes communications through alternative paths to optimize performance and reliability. Flighter models the impact on communication latency rather than physical movement to simulate mobility. Changes in the relative positions of aerial vehicles influence communication stability, with variations in latency corresponding to distances between nodes.

The defense module mitigates the risk posed by malicious nodes attempting to introduce poisoned models or engage in adversarial behaviors within the federation. It ensures the integrity and security of the models trained by each aerial vehicle. This is achieved by evaluating key criteria, including the following.

Model Similarity: Each vehicle periodically sends model updates, and the defense module performs a similarity check to compare these updates with the aggregated model from previous rounds and other participants' updates. The module calculates the deviation of the new model parameters from the expected trajectory of model learning. Any significant deviation from the expected model behavior can indicate a potential poisoning attack.

Situational Awareness: The module evaluates the operational integrity of each vehicle using various situational awareness metrics, including:

- *Flight Formation.* The module ensures that each aerial vehicle maintains its designated formation by continuously monitoring standardized position to the formation scheme.
- *Geopositioning.* Real-time geographical location data is analyzed using the Euclidean distance between each vehicle position and its planned trajectory. Significant deviations

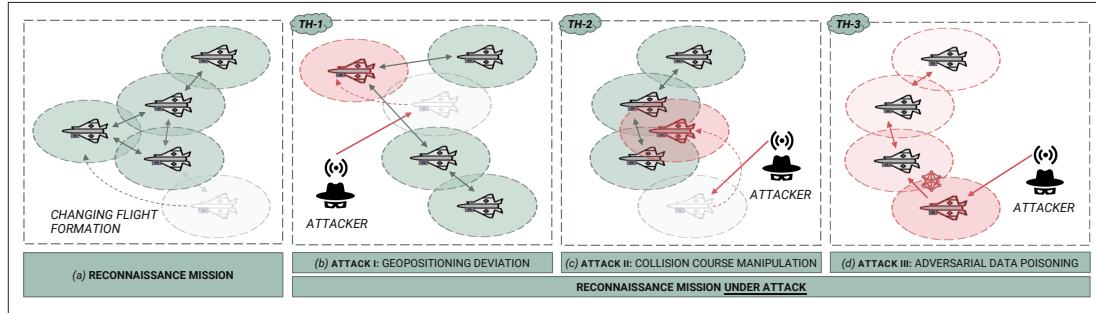


FIGURE 3. Illustration of various missions deployed during the validation of Flighter, aligned with the threats defined in this work.

- could indicate a navigation system tampering.
- *Interconnection.* The reliability and frequency of vehicle communications are evaluated based on the communication latency. Irregularities, such as dropped or delayed connections, could suggest attempts to disrupt the network.
- *Resource Usage.* The module tracks the computational resources consumed by each aerial vehicle. Unusually high or low resource usage can signal malicious activity or attempts to interfere with the training process.

After collecting the metrics, a defense score for each aerial vehicle is calculated using a three-step process (Fig. 2):

- Metrics are standardized to a common scale [0–1] for consistency.
- Weights are applied to each metric, with model similarity receiving the highest priority (30 % of the total score) due to its critical role in detecting potential poisoning attacks, without compromising performance in federations with heterogeneous data. The remaining 70% is distributed among situational awareness metrics based on their importance to the operational context: flight formation (30 %), geopositioning (15 %), interconnection (15 %), and resource usage (10 %).
- A weighted sum of these metrics is calculated, resulting in the defense score, which reflects the overall vehicle reliability within the federation. These weightings can be adjusted to the specific use case and operational requirements.

This defense score is then compared against a dynamic threshold that adapts based on training conditions. In the early rounds (first 20% of federated rounds), model updates are more variable due to noise and uncertainty, so the regularization term accounts for these fluctuations, and the threshold is set lower to avoid prematurely flagging vehicles as malicious. As the model converges and regularization penalizes deviations more, the threshold gradually increases to 0.9, ensuring that later-stage deviations are more likely to indicate malicious behavior. As shown in 2, vehicles with defense scores above the dynamic threshold are classified as benign, receiving increased weight in the aggregation process to prioritize trusted inputs ⑥. Those below the threshold are flagged as potentially malicious, and their contribution weights in the aggregation are reduced. If this behavior continues for three or more rounds, their connections are severed to protect the integ-

rity of the learning process ⑦.

The simulation and threat modules are essential for evaluating Flighter, setting the battlefield scenario. The simulation module enables the creation of realistic and dynamic aerial military scenarios. It deploys simulated military aerial vehicles using Docker containers, emulating the behavior of individual aircraft. The simulation also encompasses network and path conditions, providing a comprehensive network environment that models bandwidth, delay, and loss parameters. At the same time, it manages standard military formation patterns like Diamond and Right Hand Finger Four [13], with standardized distances between aerial vehicles to ensure optimal communication and coordination. The threat module can introduce perturbations to the longitude, latitude, and altitude of an aerial vehicle, causing it to deviate from the formation or move closer to other vehicles. The module also simulates adversarial attacks on the federated model by injecting false information into the local models, mimicking data poisoning and integrity attacks. These attacks are based on the threat model of the second section.

FLIGHTER PERFORMANCE IN A MILITARY RECONNAISSANCE MISSION UNDER ATTACK

Four missions have been defined to reproduce the threats described earlier: one reconnaissance mission without attacks and three with different attacks (geopositioning deviation, collision course manipulation, and adversarial data poisoning), as detailed in Fig. 3. Attacks were generated by an adversary targeting one aircraft in round 20, deployed through the threat module of Flighter.

RECONNAISSANCE MISSION

A simulation of four aircraft was deployed using Docker containers, which provided an ideal solution for replicating the interaction between aircraft due to their ability to ensure resource isolation, inter-node communication, and efficient management of system resources. The host system had an Intel i7-10700F CPU and 100 GB RAM, with each aircraft being allocated to a different core to mimic realistic processing capabilities. Docker flexibility and scalability enabled seamless integration of key simulation components, such as geolocation tracking, movement paths, and attack scenarios, ensuring accurate modeling of complex aerial operations and adversarial interactions. Network conditions were simulated to reflect the

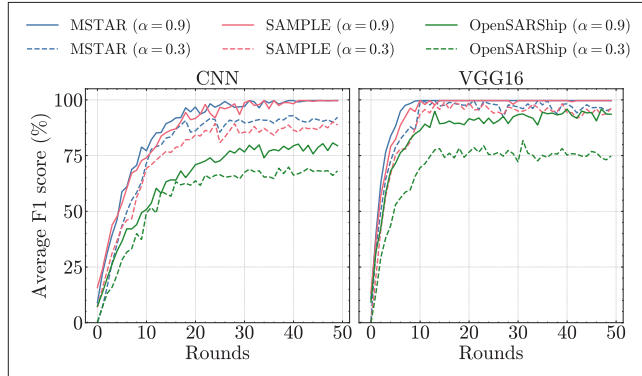


FIGURE 4. Comparison of model performance using Flighter without attacks.

Characteristic	Description
Aerial Reconnaissance Vehicles	• Four simulated military aircraft (Docker containers) • Equipped with Flighter framework
Air Formation	• Diamond and Right Hand Finger Four • 1 mile laterally, 100 feet vertically • Dynamic movement every 5 min
Dataset	• Non-IID data (Dirichlet $\alpha = 0.3$) • MSTAR and SAMPLE with ten military terrestrial vehicles • OpenSARShip with six maritime vehicles
Federated Model	• CNN-based ReLU and VGG16 • 50 federated rounds
Communications	• 2–3 MB/s bandwidth with a 10 % message loss • Delay of 5–10 ms • Transmit geopositioning data ten times per second
Vector Attacks	① 10–15° in latitude/longitude, 50–200 meters in altitude ② 200–250 meters toward its neighbors (gradual) ③ • Carlini Wagner (0.9 C, 0.01 α, 1000 iterations) • Gradient deviation attack (mean-based) [14] • Backdoor pixel trigger (top-left “X,” random target)

TABLE 1. Military aircraft configuration during the missions.

challenges of air-based communication. Each aircraft was allocated 3 MB/s bandwidth using the tc wrapper on the communication interface. Additionally, 5–10 ms network delays were introduced to replicate real message exchange over the air.

Military aircraft were equipped with a simulated SAR system to detect enemy military vehicles. SAR technology provides high-resolution remote sensing images regardless of weather, time, or light conditions. The mission utilized three datasets to evaluate the effectiveness of Flighter. First, the MSTAR dataset, developed by the Air Force Research Laboratory and DARPA, includes SAR images of various military vehicles such as tanks, personnel carriers, and air units. Second, the SAMPLE dataset provides measured and synthetic image pairs for ten public MSTAR target classes, extending the training and evaluation diversity. Finally, the OpenSARShip dataset, consisting of SAR ship chips collected from 41 C-band Sentinel-1 images, contains ship types such as tankers, container ships, and bulk carriers to assess performance in maritime scenarios [10].

The data was distributed as Non-Independent and Identically Distributed (Non-IID) using a Dirichlet distribution with $\alpha = 0.3$ and $\alpha = 0.9$, simulating varying levels of data heterogeneity across

aircraft. With $\alpha = 0.3$, data mimics real-world conditions where some aircraft primarily detect specific classes, which results in around 20% of classes being underrepresented. In contrast, $\alpha = 0.9$ offers a more balanced class distribution better suited for uniform operational contexts. For the rest of the experiments, $\alpha = 0.3$ was chosen for its closer alignment with heterogeneous data scenarios. Each aircraft deployed a lightweight Convolutional Neural Network (CNN), consisting of four convolutional blocks with ReLU activations and max-pooling, followed by a dropout layer. In addition to the CNN, a VGG16 architecture was employed, which features a significantly deeper network with 16 layers, primarily composed of convolutional layers, followed by fully connected layers for classification. VGG16 is known for its effectiveness in capturing complex image patterns and has been widely used in SAR image classification tasks. Both models were trained over 20 epochs using Stochastic Gradient Descent (SGD) for hyperparameter tuning. Figure 4 shows the baseline F1 score of Flighter across previous models and datasets under each a distribution.

During the reconnaissance mission, aircraft formations alternated every 5 minutes between Diamond and Right Hand Finger Four, maintaining 1-mile lateral and 100-foot vertical separations. Each aircraft began with a standardized position relative to the formation scheme. The simulation module continuously monitored these distances to ensure correct formation behavior, which was detected as normal by the situational awareness metrics. The federation process involved 50 rounds. This number of rounds was chosen based on our experiments, which demonstrated optimal model convergence at that point. Table 1 details the configuration and specifications of the military aircraft deployed during the mission.

ATTACK I: GEOPOSITIONING DEVIATION

In this scenario, an external attacker affects the geopositioning module of an aircraft by introducing perturbations to its latitude, longitude, and altitude data, aligning with TH-1 (Fig. 3b). This caused the aircraft to deviate from its intended flight path, leading to isolation from the formation and compromising coordination and communication with other aircraft. From the DFL perspective, this isolation means the knowledge and data from the affected aircraft, each holding different subsets of the dataset, were excluded from the model updates. This exclusion results in longer round times due to prolonged data reception waits and reduces the overall accuracy of the learning process. Consequently, the attack disrupted the physical formation and degraded the collective intelligence and operational efficiency of the entire squadron, compromising mission success and resilience.

ATTACK II: COLLISION COURSE MANIPULATION

In the second scenario, the attack manipulated the trajectory data of an aircraft to move it dangerously close to other aircraft, aligning with TH-2 (Fig. 3c). By introducing deviations that altered the course of the aircraft by 200 to 250 meters toward its neighbors, the attack created a collision course. This manipulation increased the risk of mid-air collisions and disrupted formation cohesion, threatening the safety of the aircraft and aiming to create

Military Aerial Mission	Model Performance (F1 score) ↑			Resource Usage				Round Time (s) ↓
	MSTAR	SAMPLE	OpenSARShip	CPU (%) ↓	RAM (GB) ↓	Bandwidth (MB/s) ↑	Packet Loss (%) ↓	
Fighter (w/ defense module)								
Reconnaissance (baseline)	95.8 ± 0.2	97.5 ± 0.3	79.1 ± 0.5	57 ± 4	1.7 ± 0.2	2.6 ± 0.1	0.7 ± 0.1	5.3 ± 0.3
Geopositioning Deviation	91.8 ± 2.0	90.3 ± 1.8	72.0 ± 1.5	67 ± 3	1.8 ± 0.2	2.7 ± 0.1	0.7 ± 0.2	5.4 ± 0.2
Collision Course Manipulation	92.5 ± 1.9	89.8 ± 2.1	70.1 ± 2.2	65 ± 2	1.7 ± 0.2	2.5 ± 0.2	0.8 ± 0.3	5.4 ± 0.1
Adversarial Data Poisoning								
ADP-1 (Carlini & Wagner)	90.1 ± 2.5	88.7 ± 2.3	65.0 ± 1.9	67 ± 4	2.2 ± 0.2	2.8 ± 0.2	0.7 ± 0.1	5.6 ± 0.3
ADP-2 (Gradient deviation [14])	71.3 ± 2.9	72.9 ± 2.0	52.1 ± 1.4	75 ± 3	2.2 ± 0.3	2.9 ± 0.2	0.8 ± 0.1	5.6 ± 0.5
ADP-3 (Backdoor pixel trigger)	85.0 ± 2.0	83.5 ± 1.7	58.8 ± 1.9	73 ± 8	2.2 ± 0.2	2.9 ± 0.2	0.9 ± 0.1	5.6 ± 0.3
State-of-the-art DFL Framework (w/o defense module)								
Reconnaissance (baseline)	98.1 ± 0.2	95.5 ± 0.3	78.1 ± 0.5	45 ± 3	1.5 ± 0.2	2.5 ± 0.1	0.7 ± 0.1	5.2 ± 0.3
Geopositioning Deviation	72.3 ± 2.5	70.2 ± 2.3	55.1 ± 1.5	61 ± 3	1.8 ± 0.2	2.7 ± 0.1	0.9 ± 0.2	5.4 ± 0.3
Collision Course Manipulation†	43.3 ± 1.6	41.0 ± 1.7	30.1 ± 1.2	65 ± 2	1.7 ± 0.2	2.5 ± 0.2	0.9 ± 0.6	5.4 ± 0.1
Adversarial Data Poisoning								
ADP-1 (Carlini & Wagner)	18.5 ± 2.3	17.5 ± 2.0	16.3 ± 1.9	70 ± 8	2.2 ± 0.2	2.8 ± 0.2	1.1 ± 0.6	5.6 ± 0.3
ADP-2 (Gradient deviation [14])	15.0 ± 2.5	18.9 ± 4.7	12.0 ± 2.4	75 ± 4	2.8 ± 0.2	2.4 ± 1.2	0.8 ± 0.1	6.1 ± 0.7
ADP-3 (Backdoor pixel trigger)	25.1 ± 3.4	22.3 ± 2.8	19.8 ± 2.5	86 ± 7	3.4 ± 0.6	2.9 ± 1.2	0.9 ± 0.1	5.9 ± 0.6
† Performance in case of loss of model knowledge when colliding with another military aircraft.								

† Performance in case of loss of model knowledge when colliding with another military aircraft.

TABLE 2. Average performance metrics of Flighter for the reconnaissance mission under different attacks using VGG16 across three datasets (MSTAR, SAMPLE, OpenSARShip).

confusion and hinder operational effectiveness. From the DFL perspective, erratic maneuvers or sudden evasive actions would reduce the reliability of the model updates provided by the affected aircraft. Additionally, adjusting formations or increasing distances between aircraft to accommodate these movements would extend convergence time due to increased communication latency.

ATTACK III: ADVERSARIAL DATA POISONING

The third scenario involved three distinct adversarial attack scenarios to evaluate the robustness of the defense module, aligning with TH-3 (Fig. 3d). ADP-1 employed an evasion attack inspired by the Carlini & Wagner method. This attack injected subtle perturbations into SAR image data using a confidence score of 0.9, a learning rate of 0.01, and up to 1000 iterations, resulting in misclassifications such as identifying an enemy tank as a medical evacuation truck, thereby degrading the performance of the federated model. The parameters were selected empirically to achieve a non-targeted reduction of more than 50% in model performance with minimal perturbation. ADP-2 was based on [14], which manipulates gradients by introducing controlled deviations derived from the mean and standard deviation of benign updates, allowing the adversary to influence the model within plausible parameter bounds and without full access to the gradients. ADP-3 involved a backdoor attack, where an “X” pixel pattern was introduced in the top-left corner of random targeted images. This embedded trigger enables malicious models to misclassify images containing the trigger (e.g., enemy vehicles) while behaving normally otherwise, making it difficult to detect through conventional monitoring methods.

FLIGHTER PERFORMANCE

Table 2 compares Flighter performance across four previous scenarios (normal and under attack)

using the VGG16 model and datasets from above. Additionally, it is compared against a state-of-the-art DFL framework, such as Fedstellar [15], which mirrors the training process, model architecture, data processing, and federation rounds of Flighter. However, the state-of-the-art DFL framework lacks the situational awareness-based defense module that enables Flighter to make adaptive aggregation adjustments and respond in real time to cyberattacks. Key metrics, including federated model performance, resource usage, and round time, are included with arrows (↑ for favorable, ↓ for unfavorable) to indicate optimal performance directions.

In the baseline reconnaissance mission without attacks, Flighter achieved high model performance, with F1 scores of 95.8–97.5% on the MSTAR and SAMPLE datasets and 79.1% on OpenSARShip. The defense module introduced minimal overhead, with CPU usage increasing from 45% in the baseline framework to 57% in Flighter. The impact on network bandwidth and round times was negligible, indicating that the defense module provided by Flighter does not compromise performance under normal conditions.

During the missions under attack, Flighter successfully countered the three defined attacks. In the geopositioning deviation scenario, it maintained F1 scores around 90–92% on MSTAR and SAMPLE and 72.0% on OpenSARShip, with CPU usage rising moderately to 67%. Network efficiency remained stable, with packet loss below 1% and only slight increases in round times. In contrast, the performance of the baseline framework dropped significantly, with F1 scores falling to 70.2–72.3% on MSTAR and SAMPLE and 55.1% on OpenSARShip, highlighting its inability to mitigate the impact of the compromised aircraft.

Under the collision course manipulation attack, Flighter maintained F1 scores above 89% on MSTAR and SAMPLE and 70.1% on Open-

Flighter was validated in a simulated reconnaissance mission with four military aircraft with SAR systems to identify enemy vehicles using MSTAR, SAMPLE, and OpenSarShip datasets. The mission faced geopositioning deviation affecting location accuracy, collision course manipulation compromising flight paths, and adversarial data poisoning attacks targeting the integrity of model data. The results showed significant improvements over the baseline, with F1 scores in MSTAR of 91.8% during geopositioning deviation, 92.5% under collision course manipulation, and up to 90.1% when countering adversarial data poisoning, while resource usage remained similar to the mission without attacks.

SARShip, efficiently identifying and isolating the compromised aircraft without overloading system resources (CPU usage at 65%). Conversely, the baseline framework suffered severe degradation, with F1 scores dropping below 43.3% on MSTAR and SAMPLE and 30.1% on OpenSarShip, unable to prevent collisions or manage communication disruptions. Packet loss increased to nearly 5%, and round times were extended significantly.

Finally, in the adversarial data poisoning attack, the defense mechanisms provided by Flighter were particularly effective. Despite the attacks, Flighter maintained F1 scores between 52.1–90.1%, depending on the attack type and dataset, by detecting and filtering out malicious data. Notably, in the gradient deviation and backdoor attack on the OpenSarShip dataset, the F1 score decreased to 52.1% and 58.8%, indicating a significant impact but still outperforming the baseline framework. Resource usage increased due to the computational demands of the defense module, with CPU usage reaching up to 75% but remaining within acceptable limits. The baseline framework experienced drastic declines in model performance, with F1 scores plummeting to as low as 15% under certain attacks like the gradient deviation. This highlights its vulnerability and the severe impact of malicious data on model performance without adequate protection.

CONCLUSIONS

This work presents Flighter, a framework that leverages a defense module based on situational awareness to enhance the performance of DFL in military missions under attack. The module assigns a defense score to each vehicle based on model similarity and situational awareness metrics, prioritizing inputs from reliable sources and mitigating potential threats. Flighter was validated in a simulated reconnaissance mission with four military aircraft with SAR systems to identify enemy vehicles using MSTAR, SAMPLE, and OpenSarShip datasets. The mission faced geopositioning deviation affecting location accuracy, collision course manipulation compromising flight paths, and adversarial data poisoning attacks targeting the integrity of model data. The results showed significant improvements over the baseline, with F1 scores in MSTAR of 91.8% during geopositioning deviation, 92.5% under collision course manipulation, and up to 90.1% when countering adversarial data poisoning, while resource usage remained similar to the mission without attacks.

In future work, advanced aggregation algorithms will be explored to enhance security and efficiency. Also, techniques such as knowledge distillation, pruning, quantization, prototype learning, and novel compression methods could reduce communication overhead while preserving DFL advantages. Finally, the adaptation of Flighter to multi-domain operations, including aerial, ground, and naval units, along with integrating realistic motion models, will solidify its role in next-generation military operations and communications.

ACKNOWLEDGMENT

This work has been partially supported by 21629/FPI/21, Fundación Séneca, Región de Murcia (Spain); CDL-TALENTUM from the Spanish National Institute of Cybersecurity (INCIBE) by the Recovery, Transformation, and Resilience Plan, Next Generation EU; the Swiss Federal Office for

Defense Procurement (armasuisse) with the DECIMAL project; and the University of Zürich.

REFERENCES

- [1] D. Qi et al., "Autonomous Reconnaissance and Attack Test of Uav Swarm Based on Mosaic Warfare Thought," *Proc. Int'l. Conf. Robotics and Automation Engineering*, 2021, pp. 79–83.
- [2] A. Sumari, K. Widyasari, and V. Lestari, "A Knowledge Growing Systembased Decision Making-Support System Application for Forces Command and Control in Military Operations," *Proc. Int'l. Conf. ICT for Smart Society*, 2021, pp. 1–6.
- [3] E. T. M. Beltrán et al., "Decentralized Federated Learning: Fundamentals, State of the Art, Frameworks, Trends, and Challenges," *IEEE Commun. Surveys & Tutorials*, vol. 25, no. 4, 2023, pp. 2983–3013.
- [4] A. A. Khalil and M. Ashiqur Rahman, "Fed-Up: Federated Deep Reinforcement Learning-Based UAV Path Planning Against Hostile Defense System," *Proc. Int'l. Conf. Network and Service Management*, 2022, pp. 268–74.
- [5] L. Lin and X. Zhang, "PPVerifier: A Privacy-Preserving and Verifiable Federated Learning Method in Cloud-Edge Collaborative Computing Environment," *IEEE Internet of Things J.*, 2023, pp. 8878–92.
- [6] Y. Qu et al., "Decentralized Federated Learning for Uav Networks: Architecture, Challenges, and Opportunities," *IEEE Network*, vol. 35, no. 6, 2021, pp. 156–62.
- [7] Y. Cheriguene et al., "Towards Reliable Participation in Uav-Enabled Federated Edge Learning on Non-lid Data," *IEEE Open J. Vehicular Technology*, vol. 5, 2023, pp. 125–41.
- [8] B. Brik, A. Ksentini, and M. Bouaziz, "Federated Learning for Uav-enabled Wireless Networks: Use Cases, Challenges, and Open Problems," *IEEE Access*, vol. 8, 2020, pp. 53,841–49.
- [9] C. Feng et al., "Blockchain-Empowered Decentralized Horizontal Federated Learning for 5G-Enabled UAVs," *IEEE Trans. Industrial Informatics*, 2022, p. 3582.
- [10] E. T. M. Beltrán, "CyberDataLab/flighter," 07, 2024; available: <https://github.com/CyberDataLab/flighter>.
- [11] Yang et al., "Dynamic Personalized Federated Learning With Adaptive Differential Privacy," *Proc. Int'l. Conf. Neural Information Processing Systems*, 2024.
- [12] A. Parthasarathy and B. Krishnamachari, "DEFER: Distributed Edge Inference for Deep Neural Networks," *Proc. Int'l. Conf. COMMunication Systems & NETWORKS*, 2022, p. 749.
- [13] Formation and S. T. (FAST), "Fundamentals of Formation for All Aircraft Based on Fast Formation Protocols," 12, 2016; available: <http://flyfast.org/content/reference-material>.
- [14] M. Fang et al., "Local Model Poisoning Attacks to Byzantine-Robust Federated Learning," *Proc. USENIX Conf. Security Symposium*, 2020.
- [15] E. T. M. Beltrán et al., "Fedstellar: A Platform for Decentralized Federated Learning," *Expert Systems with Applications*, vol. 242, 2024, p. 122861.

BIOGRAPHIES

ENRIQUE TOMÁS MARTÍNEZ BELTRÁN is working toward a Ph.D. in Computer Science at the University of Murcia, Spain. His research interests include cybersecurity, IoT, and collaborative learning using FL.

PEDRO M. SÁNCHEZ SÁNCHEZ is a senior researcher at the University of Murcia. He received a PhD in Computer Science from the University of Murcia, Spain. His research interests focus on ML, 5G, and cybersecurity.

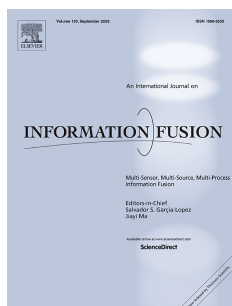
GÉRÔME BOVET is the head of data science for the Swiss Department of Defense. He received his Ph.D. in networks and computer systems from Telecom ParisTech, France. His work focuses on ML, emphasizing anomaly detection and collaborative learning in IoT sensors.

BURKHARD STILLER chairs the Communication Systems Group CSG, Department of Informatics IfI, University of Zürich, as a Full Professor. He received his Ph.D. from the University of Karlsruhe, Germany. His research interests include decentralized systems, including Blockchains, IoT, and telecommunication economics.

GREGORIO MARTÍNEZ PÉREZ is a Full Professor at the Department of Information and Communications Engineering of the University of Murcia, Spain. His scientific activity is mainly devoted to cybersecurity and networking, where he has published 160+ papers.

ALBERTO HUERTAS CELDRÁN is an assistant professor at the University of Murcia and a guest researcher at the Communication Systems Group CSG, Department of Informatics IfI, University of Zürich. He received his PhD in Computer Science from the University of Murcia, Spain. His scientific interests include cybersecurity, FL, and computer networks.

Decentralized Federated Learning with Multimodal Prototypes for Heterogeneous Data



Title:	Decentralized Federated Learning with Multimodal Prototypes for Heterogeneous Data
Authors:	Enrique Tomás Martínez Beltrán, G�r�me Bovet, Gregorio Mart�nez P�rez, Alberto Huertas Celdr�n.
Journal:	Information Fusion
JIF:	15.5 Q1-D1 (2024)
Publisher:	Elsevier
Volume:	132
Article number:	104251
Month:	Feb
Year:	2026
DOI:	10.1016/j.inffus.2026.104251
Status:	Published

Abstract

Decentralized Federated Learning (DFL) enables collaborative machine learning across numerous devices while avoiding bottlenecks and reliance on a single trusted entity inherent to centralized architectures. However, its practical application is challenged by modern scenarios where data is increasingly multimodal. The key obstacles in such settings are severe data heterogeneity, characterized by non-Independent and Identically Distributed (non-IID) class distributions, and incomplete data, where modalities are often missing across clients. Existing solutions struggle with these challenges, either incurring high communication costs or lacking effective mechanisms for fusing partial information. To overcome these limitations, this work introduces Modalis, a novel framework for multimodal DFL that achieves superior model performance under data heterogeneity while minimizing network consumption. It pioneers a communication-efficient, prototype-centric protocol in which clients exchange compact, modality-aware class representations rather than high-dimensional model parameters. This process is guided by a multi-objective loss function enforcing inter-modality coherence and representation alignment for effective knowledge fusion. The framework integrates sophisticated architectural innovations, including contextual null embeddings for intelligent data imputation and robust multimodal fusion using adaptive gating and multi-way transformers. The approach is validated through theoretical analysis, providing formal convergence guarantees, and extensive experiments on standard multimodal benchmarks. These results demonstrate that Modalis achieves superior performance,

improving F1 scores by up to 4% under high heterogeneity and reducing communication costs by over 40 times compared to state-of-the-art baselines, establishing it as a highly effective solution for collaborative AI.

Keywords

Federated Learning · Multimodal Learning · Prototype Learning · Data Heterogeneity
· Missing Modalities · Communication Efficiency



Contents lists available at ScienceDirect

Information Fusion

journal homepage: www.elsevier.com/locate/infus

Full Length Article

Decentralized Federated Learning with Multimodal Prototypes for Heterogeneous Data

Enrique Tomás Martínez Beltrán ^{a,*}, G  r  me Bovet ^b, Gregorio Mart  nez P  rez ^a,
Alberto Huertas Celdr  n ^a

^a Department of Information and Communications Engineering, University of Murcia, Murcia, 30100, Spain

^b Cyber-Defence Campus, Armasuisse Science and Technology, Thun, 3602, Switzerland

ARTICLE INFO

Keywords:

Federated learning
Multimodal learning
Prototype learning
Data heterogeneity
Missing modalities
Communication efficiency

ABSTRACT

Decentralized Federated Learning (DFL) enables collaborative machine learning across numerous devices while avoiding bottlenecks and reliance on a single trusted entity inherent to centralized architectures. However, its practical application is challenged by modern scenarios where data is increasingly multimodal. The key obstacles in such settings are severe data heterogeneity, characterized by non-Independent and Identically Distributed (non-IID) class distributions, and incomplete data, where modalities are often missing across clients. Existing solutions struggle with these challenges, either incurring high communication costs or lacking effective mechanisms for fusing partial information. To overcome these limitations, this work introduces MODALIS, a novel framework for multimodal DFL that achieves superior model performance under data heterogeneity while minimizing network consumption. It pioneers a communication-efficient, prototype-centric protocol in which clients exchange compact, modality-aware class representations rather than high-dimensional model parameters. This process is guided by a multi-objective loss function enforcing inter-modality coherence and representation alignment for effective knowledge fusion. The framework integrates sophisticated architectural innovations, including contextual null embeddings for intelligent data imputation and robust multimodal fusion using adaptive gating and multi-way transformers. The approach is validated through theoretical analysis, providing formal convergence guarantees, and extensive experiments on standard multimodal benchmarks. These results demonstrate that MODALIS achieves superior performance, improving F1 scores by up to 4% under high heterogeneity and reducing communication costs by over 40 times compared to state-of-the-art baselines, establishing it as a highly effective solution for collaborative AI.

1. Introduction

Federated Learning (FL) [1] has emerged as a paradigm-shifting approach that enables collaborative machine learning on distributed datasets without centralizing potentially sensitive user data. This privacy-preserving characteristic is crucial for applications in health-care, finance, and personal devices. However, the conventional server-client topology inherent to most FL systems introduces potential bottlenecks and single points of failure [2]. Decentralized Federated Learning (DFL) [3] mitigates these issues by facilitating direct peer-to-peer communication and model aggregation among clients, thereby enhancing system robustness, scalability, and eliminating reliance on a central trusted entity. Despite these advantages, achieving efficient knowledge aggregation and stable convergence in DFL remains a formidable

challenge, particularly given the multifaceted complexities of real-world data distributions.

Real-world applications rarely rely on a single data source; instead, they typically involve diverse modalities such as images, text, audio, and structured sensor readings. A significant and often overlooked challenge arises from the *multimodality* and *heterogeneity* of data dispersed across participating clients [4]. In decentralized networks, clients frequently contend with *missing modalities* (e.g., a client possesses text data but lacks the corresponding image for a given sample) and highly skewed, non-Independent and Identically Distributed (non-IID) class distributions [5]. Such multifaceted heterogeneity poses a fundamental challenge to existing FL paradigms. Conventional algorithms that rely on parameter averaging are particularly vulnerable; averaging parameters from a client's untrained sub-network, a direct consequence of a missing

* Corresponding author.

E-mail addresses: enriquetomas@um.es (E.T. Mart  nez Beltr  n), gerome.bovet@armasuisse.ch (G. Bovet), gregorio@um.es (G. Mart  nez P  rez), alberto.huertas@um.es (A. Huertas Celdr  n).

<https://doi.org/10.1016/j.inffus.2026.104251>

Received 2 August 2025; Received in revised form 26 November 2025; Accepted 18 February 2026

Available online 20 February 2026

1566-2535/   2026 The Author(s). Published by Elsevier B.V. This is an open access article under the CC BY-NC license (<http://creativecommons.org/licenses/by-nc/4.0/>).

modality, inevitably pollutes the aggregated model and nullifies peer contributions. While more advanced frameworks offer a solution for non-IID data by exchanging compact, class-level feature embeddings, known as prototypes, instead of full model parameters, they were predominantly designed for unimodal tasks and thus lack native mechanisms for multimodal fusion. Finally, the few solutions that address multimodal heterogeneity almost universally rely on a centralized topology for complex operations, such as fusion or imputation, thereby reintroducing the very bottlenecks and single points of failure that DFL is designed to eliminate.

To bridge this critical gap, this paper introduces MODALIS, a novel framework designed for efficient and robust multimodal learning in heterogeneous, decentralized environments. The cornerstone of MODALIS is its *prototype-centric communication protocol*. Instead of transmitting large, high-dimensional model parameters or gradients, clients exchange only low-dimensional *class prototypes*. These prototypes are learnable embeddings that represent the salient, class-discriminative features for specific modalities, as learned by a client from its local data. This strategy achieves substantial communication savings, formalized in [Lemma 1](#), and offers potential privacy benefits by sharing abstract, aggregated representations rather than raw model updates, which can be vulnerable to sophisticated inference attacks [\[6\]](#).

More in detail, this work provides a comprehensive set of technical contributions to advance the state of the art in DFL:

- **Modular framework:** MODALIS is designed and implemented, featuring a novel prototype-based communication protocol at its core. It incorporates several architectural innovations: a Contextual Null Embedding module for intelligent, feature-conditioned imputation of missing modalities; an adaptive gating mechanism that weights modality contributions based on their reliability; and a masked multi-way transformer for robust information fusion that explicitly handles missing-data patterns.
- **Theoretical guarantees:** The work is supported by a rigorous theoretical analysis, which includes a formal convergence proof establishing that MODALIS is guaranteed to reach a stationary neighborhood of an optimal solution ([Theorem 1](#)). Additionally, a detailed lemma quantifies the significant reduction in communication cost of the framework compared to traditional model-passing DFL methods ([Lemma 1](#)).
- **Rigorous validation setup:** An extensive experimental protocol is designed to validate the framework performance. The evaluation is conducted on four diverse and widely recognized multimodal datasets (AVMNIST, CREMA-D, UCI-HAR, CRISIS-MMD). Realistic data heterogeneity is simulated by systematically varying the degree of non-IID class skew (controlled by Dirichlet parameter $\beta \in \{1.0, 0.5, 0.1\}$) and the proportion of missing modalities across clients (controlled by $p_m \in \{0.0, 0.25, 0.5\}$).
- **State-of-the-art results:** The empirical results show that MODALIS consistently outperforms established FL and DFL baselines. Under high-heterogeneity conditions, it achieves a relative F1 score gain of up to $\approx 4.0\%$ over the next-best competing method. The prototype-based communication reduces network overhead by over **40x** compared to methods that exchange full model parameters. Furthermore, its strategy of performing only a single local training epoch ($T = 1$) per communication round minimizes the computational burden on clients and effectively mitigates model drift.

The remainder of this paper is organized as follows: [Section 2](#) reviews pertinent literature in federated and decentralized learning. [Section 3](#) formally defines the problem setup and notation. [Section 4](#) provides a detailed exposition of the MODALIS framework, beginning with its core components and concluding with its operational protocol. [Section 5](#) discusses the theoretical properties of the proposed method. [Section 6](#) presents the comprehensive experimental evaluation. Finally, [Section 8](#) concludes the paper with a summary of findings and a discussion of promising directions for future research.

2. Related work

This section reviews the literature on prototype-based and multimodal FL, as well as on techniques for addressing data heterogeneity, thereby contextualizing the contributions of the work.

2.1. Prototype-based federated learning

Prototype-based FL methods have emerged as a compelling strategy to mitigate high communication costs and improve model robustness against non-IID data. These methods work by exchanging compact, class-representative embeddings (prototypes) rather than voluminous, potentially sensitive model parameters. In the context of centralized FL, the work on FedProto by Tan et al. [\[7\]](#) was seminal. It demonstrated that by having clients exchange only class prototypes with a central server, both communication costs and the negative effects of gradient misalignment under non-IID data could be significantly reduced. The global prototypes shared by the server act as a regularizer, guiding local training. Building on this, FedProc [\[8\]](#) introduced a prototypical contrastive regularizer that forces each client's local feature distributions to align more closely with the globally shared class prototypes, reporting accuracy improvements of $1.6 - 7.9\%$ over standard FedAvg by correcting local training drift on skewed image classification tasks. Similarly, Tran et al. [\[9\]](#) proposed FedNTPProto, a personalized FL approach that uses class-specific prototypes to tackle client heterogeneity. By sharing and distilling knowledge through nearest-class prototype representations, each client can retain a personalized model while benefiting from insights into the data distributions of other clients, thereby improving local performance on heterogeneous vision tasks.

The concept of prototype exchange has also been explored in DFL. Qi et al. [\[10\]](#) developed PearFL, a peer-to-peer FL system in which neighboring clients periodically exchange lightweight prototypes rather than full models. This decentralized prototype-sharing mechanism was shown to improve communication efficiency and model consistency across edge devices, with experiments indicating faster convergence and higher accuracy compared to centralized FL in resource-constrained, heterogeneous networks. Li et al. [\[11\]](#) proposed DeProFL, a DFL method tailored for dynamic Internet of Things (IoT) systems that communicate prototype representations. In each round, nodes transmit only learned feature prototypes (e.g., first-layer feature vectors) to their neighbors, enabling the system to handle time-varying topologies by achieving consensus at the prototype level. DeProFL reportedly obtained comparable accuracy (within $0.5 - 5\%$ of centralized methods) while converging faster under non-convex objectives. Zhang et al. [\[12\]](#) combined blockchain technology with prototype learning to eliminate the need for a central server in FL. Their framework uses an immutable ledger to synchronize global class prototypes across clients, addressing server-failure vulnerabilities and improving communication efficiency and test accuracy across several non-IID benchmarks.

2.2. Multimodal learning in federated settings

Applying FL to multimodal data introduces unique challenges related to managing diverse data types, handling variations in modality availability across clients, and effectively and securely fusing information. Several centralized approaches have been proposed to tackle these issues. Chen and Zhang [\[13\]](#) proposed FedMBridge, which utilizes a topology-aware hypernetwork to generate “bridge” parameters, thereby enabling knowledge sharing among clients with different model architectures or modalities. Wang et al. [\[14\]](#) addressed multimodal FL by avoiding a unified multimodal model and instead training multiple single-modal models federatedly. Each client contributes to learning modality-specific feature extractors, enabling clients with limited modalities to benefit from others (e.g., in pathology, imaging-only hospitals could contribute to a federated genomics model). Rahman and Nguyen [\[15\]](#) proposed mmFL, which employs model personalization via

meta-learning for clients with heterogeneous modalities. Modality sets cluster clients, and meta-learning adapts a global model to each cluster, reportedly improving accuracy by over 7% compared to vanilla FL. Huang et al. [16] explored various fusion strategies, including early, late, and attention-based, for multimodal data in centralized FL, finding that attention-based fusion was more adaptive but costlier. These approaches, while advancing multimodal FL, primarily operate under a centralized paradigm and may not directly translate to the complexities and constraints of fully decentralized systems.

2.3. Handling data and modality heterogeneity

Addressing data heterogeneity, particularly non-IID distributions and missing modalities, is a critical prerequisite for the successful deployment of practical FL systems. Numerous specialized techniques have been developed, primarily within the centralized FL paradigm. Bao et al. [17] introduced PmcmFL, which employs prototypes as learnable placeholders for missing modalities, formulating a modality-missing loss and an unimodal inference strategy. This method reportedly improved global model performance by 3.7% when 50% of modalities were missing. Poudel et al. [18] augmented client data with similar examples from public datasets for missing modalities in their CAR-MFL method, where clients retrieve matching samples from a public repository to fill missing inputs. Others leverage more advanced machine learning techniques, such as contrastive learning and graph neural networks, to learn meaningful representations for absent modalities by aggregating information from graph neighbors [19]. An et al. [20] presented a two-step aggregation for modality-incomplete FL in healthcare: intra-cluster FL aggregation for modality-specific encoders, then inter-cluster prototype sharing and contrastive integration. Yu et al. [21] proposed FedInMM, incorporating a Long Short-Term Memory (LSTM)-based feature fusion module to handle incomplete modalities by learning temporal patterns of modality availability and dynamically adjusting feature combination. Similarly, Nguyen et al. [22] proposed CreamFL, which addresses modality heterogeneity by clustering clients based on their available modalities and data characteristics. Cluster-specific models are then trained, with a central server coordinating the learning process across the clusters. More recently, Yin et al. [23] proposed FedMAC. This centralized multimodal FL framework combines a federated self-attention fusion module with an adaptive continual update strategy and diffusion-based reconstruction of missing image modalities to improve robustness under non-IID and modality-incomplete data. In parallel, Le et al. [24] introduced MFCPL, a cross-modal prototype-based multimodal FL approach in which a central server maintains complete class prototypes in a shared representation space and uses them to regularize both modality-shared and modality-specific representations, explicitly targeting severely missing-modality settings.

As summarized in Table 1, despite these innovative advances, the vast majority of existing solutions depend on a central server for coordination, data retrieval, or model aggregation. To the best of current knowledge, no prior work comprehensively combines a fully decentralized learning protocol with a multimodal prototype fusion mechanism that is inherently capable of robustly and simultaneously handling both arbitrary missing modality scenarios and severe non-IID data distributions. The MODALIS framework presented herein is designed specifically to fill this important research gap.

3. Problem statement

This work considers a decentralized system composed of N clients, indexed by $i \in \{1, \dots, N\}$. These clients are interconnected through a communication network, which can be formally represented as an undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$. In this graph, $\mathcal{V} = \{1, \dots, N\}$ constitutes the set of clients (vertices), and $\mathcal{E}$ represents the set of available communication links (edges) between pairs of clients. A client i is constrained to exchange information directly only with its immediate neighbors, defined

by the set $\mathcal{N}_i = \{j \mid (i, j) \in \mathcal{E}\}$. The system operates in a fully decentralized manner, meaning there is no central coordinating server to orchestrate communication or aggregate models. This local-neighborhood communication model is standard in decentralized optimization and DFL, as it keeps the per-round communication cost of each client proportional only to its degree in $\mathcal{G}$, rather than to the total number of participants N . Although only immediate neighbors exchange messages at each round, information propagates multi-hop across the network through successive rounds of prototype aggregation, enabling clients to eventually benefit from data residing at distant nodes. The impact of the connectivity pattern and the efficiency of this indirect knowledge sharing are captured in the analysis in Section 5 via the mixing properties of the communication matrix and the parameter ρ .

Each client i possesses a private, local dataset denoted by $\mathcal{D}_i = \{(\mathbf{x}_i^{(s)}, y_i^{(s)}, \mathbf{m}_i^{(s)})\}_{s=1}^{n_i}$, where $n_i = |\mathcal{D}_i|$ is the number of data samples available locally on client i . Each data sample s in this dataset consists of three components:

- A multimodal input $\mathbf{x}_i^{(s)} = \{\mathbf{x}_i^{M(s)}\}_{M \in \mathcal{M}}$, where $\mathcal{M}$ is the global set of all possible modalities present within the system (e.g., $\mathcal{M} = \{\text{image, text, audio}\}$). The term $\mathbf{x}_i^{M(s)}$ denotes the data pertaining to modality M for sample s .
- A class label $y_i^{(s)} \in \{1, \dots, K\}$, where K represents the total number of distinct classes across the entire federated system.
- A modality availability mask $\mathbf{m}_i^{(s)} = \{m_i^{M(s)}\}_{M \in \mathcal{M}}$, being a binary vector where $m_i^{M(s)} = 1$ indicates that modality M is present for sample s on client i , while $m_i^{M(s)} = 0$ signifies that it is absent.

The system is defined by inherent heterogeneity, which manifests in several critical forms:

- **Data heterogeneity:** The local data distributions $P_i(\mathbf{x}, y, \mathbf{m})$ can vary significantly and arbitrarily across clients. This heterogeneity encompasses non-IID class distributions (label skew), where certain classes may be heavily over- or under-represented across clients, and feature distribution skew, where the statistical properties of features for the same class can differ across clients [2].
- **Modality heterogeneity:** Clients may possess different subsets of the global modality set $\mathcal{M}$. For instance, one client might only have image data, while another might have both image and text data. Furthermore, even for a specific client i , individual modalities can be missing for certain data samples (i.e., $m_i^{M(s)} = 0$ for some $M \in \mathcal{M}$). The pattern of missing modalities can be arbitrary and differ across both clients and samples.

The primary objective is to enable each client i to collaboratively train a powerful, personalized multimodal model parameterized by θ_i solely by interacting with its direct neighbors in a decentralized fashion. The overarching goal is to minimize a global loss function defined over the union of all local data, expressed as $F(\theta_1, \dots, \theta_N) = \sum_{i=1}^N \frac{n_i}{\sum_{j=1}^N n_j} \mathbb{E}_{(\mathbf{x}, y, \mathbf{m}) \sim \mathcal{D}_i} [\mathcal{L}_{\text{task}}(\theta_i; \mathbf{x}, y, \mathbf{m})]$. This optimization must be achieved while strictly respecting the constraints of decentralized communication and ensuring robust performance despite the aforementioned sources of systemic heterogeneity.

3.1. Notation

Table 2 provides a comprehensive summary of the principal notation employed throughout this paper to ensure clarity and consistency.

4. MODALIS Framework

The MODALIS framework provides a complete solution for decentralized multimodal learning. This section details its core elements using a top-down approach. It first outlines the high-level, round-based operational protocol that governs collaborative training. Subsequently, it

Table 1
Comparative analysis of state-of-the-art FL methods.

Method	Year	Topology	Communication Type	Multimodal	Missing Modality	Non-IID	Personalization	Domain
FedProto [7]	2022	Centralized	Prototypes	✓	✓	✓	✓	Image classification
FedProc [8]	2023	Centralized	Prototypes	✓	✓	✓	✓	Image classification
FedNTProto [9]	2024	Centralized	Prototypes	✓	✓	✓	✓	Vision tasks
DeProFL [11]	2024	Decentralized	Prototypes	✓	✓	✓	✓	IoT systems
PearFL [10]	2025	Decentralized	Prototypes	✓	✓	✓	✓	Edge networks
DFPL [12]	2025	Blockchain-based	Prototypes	✓	✓	✓	✓	Non-IID benchmarks
FedMBridge [13]	2024	Centralized	Parameters	✓	✓	✓	✓	Heterogeneous arch.
FedMMR [14]	2024	Centralized	Parameters	✓	✓	✓	✓	Missing-modality recon.
mmFL [15]	2024	Centralized	Parameters	✓	✓	✓	✓	Modality clustering
Huang et al. [16]	2024	Centralized	Parameters	✓	✓	✓	✓	Fusion strategies
PmcmFL [17]	2023	Centralized	Parameters	✓	✓	✓	✓	Prototype masking
CreamFL [22]	2023	Centralized	Parameters	✓	✓	✓	✓	Contrastive ensemble
CAR-MFL [18]	2024	Centralized	Parameters	✓	✓	✓	✓	Public-data augmentation
MIFL [19]	2024	Centralized	Parameters	✓	✓	✓	✓	GNN-based imputation
An et al. [20]	2024	Centralized	Parameters	✓	✓	✓	✓	Healthcare encoders
FedInMM [21]	2024	Centralized	Parameters	✓	✓	✓	✓	Temporal fusion
FedMAC [23]	2025	Centralized	Parameters	✓	✓	✓	✓	Self-attention fusion
MFCPL [24]	2025	Centralized	Prototypes	✓	✓	✓	✓	Severely missing modality
MODALIS (our work)	2025	Decentralized	Prototypes	✓	✓	✓	✓	General multimodal

provides a detailed exposition of the key components that enable this protocol: the client's internal architecture, the mechanisms for proto-type management, and the multi-objective loss function.

4.1. High-level protocol and operational flow

MODALIS operates through a dynamic, round-based asynchronous protocol. This protocol defines the interactions between clients and their internal computational flow, orchestrating the static components described in subsequent sections. The overall process for a client i during a single round t consists of five main steps, as illustrated in Fig. 1:

- 1. Receive prototypes:** At the start of round t , client i processes any new prototype libraries $\{\mathbf{P}_j^{t_i}\}$ that have arrived from its neighbors $j \in \mathcal{N}_i$ since its last aggregation phase. The notation t_i signifies the round in which neighbor j 's prototype was generated, accommodating stragglers and variable network latencies inherent to asynchronous DFL. The protocol does not require all neighbors to report in lockstep; instead, each client aggregates the most recent information available to it at the time of its local update.
- 2. Aggregate prototypes:** Client i computes an aggregated prototype library $\bar{\mathbf{P}}_i^t$. This aggregation combines its local prototypes from the previous round, $\mathbf{P}_i^{t-1}$, with the received neighbor prototypes using the count-weighted averaging scheme from Eq. (2).
- 3. Local training:** Client i performs exactly one epoch ($T = 1$) of training over its local dataset $\mathcal{D}_i$. During this epoch, it utilizes the freshly computed aggregated prototypes $\bar{\mathbf{P}}_i^t$ to calculate auxiliary loss components (detailed in Section 4.4) and updates its local model parameters θ_i . The choice of $T = 1$ is deliberate to minimize local computation and mitigate client drift under significant data heterogeneity [25].
- 4. Update local prototypes:** After the local training epoch, client i updates its local prototype library from $\mathbf{P}_i^{t-1}$ to $\mathbf{P}_i^t$, using accumulated feature embeddings and the adaptive EMA mechanism (detailed in §4.3). This update allows local prototypes to evolve based on newly observed data while being stabilized by their recent history.
- 5. Send prototypes:** Client i transmits its new local prototype library $\mathbf{P}_i^t$ to all its direct neighbors $j \in \mathcal{N}_i$, which disseminates its latest learned class representations to its peers for use in the next round of aggregation.

These five steps, comprising a full communication round, are formally detailed in Algorithm 1. This algorithm encapsulates the entire operational logic of a single client, from aggregation to communication.

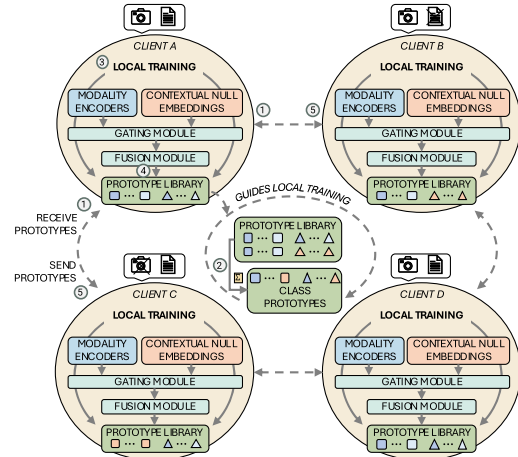


Fig. 1. MODALIS operational flow in a decentralized architecture.

4.2. Client system architecture

The local training phase of the protocol (Step 3) is executed by a sophisticated multimodal neural network, parameterized by θ_i , which is specifically designed to process multimodal inputs, handle missing data, and generate representations for both task performance and prototype generation. As illustrated in Fig. 2, the client's neural network defines the forward pass for a given sample s . Its components are executed sequentially to produce the embeddings and logits required for both final classification and subsequent multi-objective loss calculation. The key architectural components are:

- 1. Modality encoders ($\mathcal{E}_M$):** For each modality $M \in \mathcal{M}$, a dedicated encoder network $\mathcal{E}_M$ maps the raw input data $\mathbf{x}_i^{M,(s)}$ to a fixed-dimensional feature embedding $\mathbf{z}_i^{M,(s)} \in \mathbb{R}^d$. The architecture of $\mathcal{E}_M$ is chosen to be appropriate for the data type of modality M (e.g., a CNN for images, a Transformer for text).
- 2. Contextual null embedding ($\mathcal{N}_M$):** This module is a cornerstone for handling missing modalities. If modality M is absent for a sample s (i.e., $m_i^{M,(s)} = 0$), the $\mathcal{N}_M$ module generates a substitute embedding $\mathbf{z}_i^{M,(s)}$. It utilizes a learnable base null embedding vector $\mathbf{e}_{\text{null_base}}^M \in \mathbb{R}^d$.

Table 2
Summary of key notation.

Symbol	Description
B	Local mini-batch size
β	Dirichlet distribution parameter for simulating non-IID label skew
C	Final classification head network
d	Dimensionality of embedding spaces
D_i	Local dataset of client i
δ_p^2	Bound on expected squared prototype drift per round (Assumption 5)
$\mathcal{E}_M$	Encoder network for modality M
$\mathbf{e}_i^{(s)}$	Fused multimodal embedding for sample s on client i
η	Learning rate
$\mathcal{F}$	Multimodal fusion module
G^2	Bound on expected squared norm of stochastic gradients (Assumption 3)
$\mathcal{G}$	Gating module for modality embeddings
$\mathcal{G}, \mathcal{V}, \mathcal{E}$	Communication graph, vertices (clients), edges (links)
i, j	Client indices
K	Total number of classes
k	Class index, $k \in \{1, \dots, K\}$
L	Smoothness constant for local objective functions (Assumption 1)
$\mathcal{L}$	Loss function component
M	Index representing a specific modality
$\mathcal{M}$	Set of all possible modalities in the system
$m_i^{M,(s)}$	Binary mask: 1 if modality M is present for sample s on client i , 0 otherwise
m_i	EMA momentum parameter for prototype update at round t
N	Total number of clients
$\mathcal{N}_i$	Set of neighbors of client i in graph $\mathcal{G}$
n_i	Number of samples in local dataset D_i
$\mathcal{N}_M$	Contextual Null Embedding module for modality M
θ_i^t	Model parameters of client i at the end of round t
$\mathbf{P}_i^t$	Local prototype library of client i at the end of round t
$\bar{\mathbf{P}}_i^t$	Aggregated prototype library computed by client i at round t
p_m	Proportion parameter for simulating missing modalities
P_1, P_2	Projection head networks
$\mathbf{p}_i^t(*, k)$	Local prototype vector of client i for class k , type $*$ (modality/fusion), at round t
$\bar{\mathbf{p}}_i^t(*, k)$	Aggregated prototype vector computed by client i for class k , type $*$, at round t
T	Number of local training epochs per round (set to $T = 1$)
$\mathbf{r}_i^{(s)}$	Projection of fused embedding $\mathbf{e}_i^{(s)}$ via P_1 for sample s on client i
ρ	Spectral gap related parameter of the network mixing matrix (Assumption 4)
σ^2	Bound on variance of stochastic gradients (Assumption 2)
s_{DCL}	Scaling factor for $\mathcal{L}_{DCL}$
t	Communication round index
R	Total number of communication rounds
τ_{DCL}	Temperature parameter for $\mathcal{L}_{DCL}$
w	Weight hyperparameter for a loss component
$(\mathbf{x}_i^{(s)}, \mathbf{y}_i^{(s)}, \mathbf{m}_i^{(s)})$	s -th data sample on client i (multimodal input, label, modality mask)
$\mathbf{z}_i^{M,(s)}$	Feature embedding for modality M from sample s on client i
$\hat{\mathbf{z}}_i^{M,(s)}$	Gated feature embedding for modality M from sample s on client i
$\mathbf{z}_i^{M,(s)}$	Projection of modality embedding $\mathbf{z}_i^{M,(s)}$ via P_2

$\mathbb{R}^d$, which is specific to modality M . Critically, if another modality M' is present for the same sample, $\mathcal{N}_M$ conditions the generation of the null embedding on the feature representation $\mathbf{z}_i^{M',(s)}$ from the available modality. This contextual generation is achieved via an MLP:

$$\mathcal{N}_M(\mathbf{z}_i^{M',(s)}) = \text{LayerNorm} \left(\text{MLP}_{\text{ctx}}^M([\mathbf{e}_{\text{null_base}}^M; \text{StopGrad}(\mathbf{z}_i^{M',(s)})]) + \mathbf{e}_{\text{null_base}}^M \right) \quad (1)$$

where $[\cdot; \cdot]$ denotes concatenation. The use of $\text{StopGrad}(\cdot)$ is crucial as it detaches the context embedding $\mathbf{z}_i^{M',(s)}$ from the computational graph of the imputation module. It ensures that the encoder $\mathcal{E}_{M'}$ is optimized solely for extracting task-relevant features from its

Algorithm 1: MODALIS: client operation.

Input: Client i ; Current model θ_i^{t-1} ; Local dataset D_i ; Previous local prototypes $\mathbf{P}_i^{t-1}$; Neighbor prototypes $\{\mathbf{P}_j^{t-1}\}_{j \in \mathcal{N}_i}$; Hyperparameters $\{w, \cdot, \eta, T = 1, B\}$.
Output: Updated model parameters θ_i^t ; Updated local prototypes $\mathbf{P}_i^t$.

Initialize new aggregated prototype library $\bar{\mathbf{P}}_i^t$;
foreach type $*$ **and** class k **do**
 Compute $\bar{\mathbf{p}}_i^t(*, k)$ and its count $\bar{c}_i^t(*, k)$ using Eq. (2);
 Store L2-normalized $\bar{\mathbf{p}}_i^t(*, k)$ in $\bar{\mathbf{P}}_i^t$;
 $\theta_i \leftarrow \theta_i^{t-1}$ $\triangleright$ Load current model parameters Initialize feature accumulators: $\Sigma_k^* \leftarrow \mathbf{0}$, $v_k^* \leftarrow 0$ for all types $*$ and classes k ;
for batch_idx $\leftarrow 1$ **to** $\lceil n_i/B \rceil$ **do**
 Sample mini-batch $B_s = \{(\mathbf{x}_s, \mathbf{y}_s, \mathbf{m}_s)\}_{s=1}^{|B_s|}$ from D_i ;
 foreach sample $s \in B_s$ **do**
 foreach modality $M \in \mathcal{M}$ **do**
 if $m_s^M = 1$ **then**
 $\mathbf{z}_s^M \leftarrow \mathcal{E}_M(\mathbf{x}_s; \theta_i^t)$;
 else
 Select context modality M' (if available);
 $\mathbf{z}_s^M \leftarrow \mathcal{N}_M(\text{context from } \mathbf{z}_s^{M'} \text{ or null basis; } \theta_i^{\mathcal{N}_M})$;
 $\{\hat{\mathbf{z}}_s^M\}_M \leftarrow \mathcal{G}(\{\mathbf{z}_s^M\}_M; \theta_i^t)$;
 $\mathbf{e}_s \leftarrow \mathcal{F}(\{\hat{\mathbf{z}}_s^M\}_M, \mathbf{m}_s; \theta_i^t)$;
 $\mathbf{r}_s \leftarrow P_1(\mathbf{e}_s; \theta_i^t)$ and $\{\mathbf{z}_s^M\}_M \leftarrow P_2(\{\mathbf{z}_s^M\}_M; \theta_i^t)$;
 $\mathbf{z}_{\text{cls},s} \leftarrow C(\mathbf{e}_s; \theta_i^t)$;
 $L_{\text{total}} \leftarrow \sum w_{\text{loss}} \mathcal{L}_{\text{loss}}$ using Eqs. (5)-(9) on batch B_s and prototypes $\bar{\mathbf{P}}_i^t$;
 $\theta_i \leftarrow \theta_i - \eta \nabla_{\theta_i} L_{\text{total}}$;
 Update Σ_k^* and v_k^* using detached embeddings from genuine modalities in B_s ;
 $\theta_i^t \leftarrow \theta_i$;
 Initialize new local prototype library $\mathbf{P}_i^t$;
 foreach type $*$ **and** class k **do**
 if $v_k^* > 0$ **then**
 $\bar{\mathbf{z}}_k^* \leftarrow \Sigma_k^* / v_k^*$;
 Update $\mathbf{p}_i^t(*, k)$ and $\bar{c}_i^t(*, k)$ using EMA as per Eq. (3);
 else
 $\mathbf{p}_i^t(*, k) \leftarrow \mathbf{p}_i^{t-1}(*, k)$; $\bar{c}_i^t(*, k) \leftarrow \bar{c}_i^{t-1}(*, k)$;
 Store L2-normalized $\mathbf{p}_i^t(*, k)$ and $\bar{c}_i^t(*, k)$ in $\mathbf{P}_i^t$;
return $\theta_i^t, \mathbf{P}_i^t$

modality, preventing its learning objective from being distorted by the auxiliary task of imputing the missing modality M .

- Gating module (G):** This module computes attention-like reliability weights, $\alpha^{M,(s)} \in [0, 1]$, for each modality embedding $\mathbf{z}_i^{M,(s)}$ (including those imputed by $\mathcal{N}_M$). Concretely, a lightweight parametric function receives as input the concatenation of the modality embedding, a one-hot encoding of the modality index, and its availability flag, and produces unnormalized scores that are subsequently normalized with a softmax over the set of modalities present according to $\mathbf{m}_i^{(s)}$. The gated feature embeddings are then calculated as $\hat{\mathbf{z}}_i^{M,(s)} = \alpha^{M,(s)} \cdot \mathbf{z}_i^{M,(s)}$. Because these weights are learned end-to-end jointly with the task and prototype-based losses, modalities whose embeddings are consistently noisy or weakly predictive (e.g., low-quality or miscalibrated sensors) are automatically assigned smaller $\alpha^{M,(s)}$ values during training, so that their contribution to the fused representation is reduced rather than treated as equally reliable.

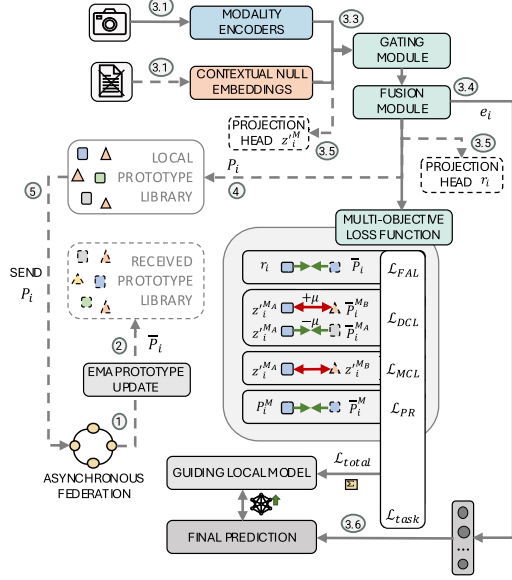


Fig. 2. MODALIS internal architecture.

4. **Multimodal fusion module (F):** A multi-way transformer architecture, inspired by [26], integrates the set of gated modality embeddings $\{\hat{z}_i^{M(s)}\}_{M \in \mathcal{M}}$ into a single, unified multimodal representation $e_i^{(s)} \in \mathbb{R}^d$. The transformer operates on these gated embeddings as a sequence of modality-specific tokens. It employs an attention mask derived from the modality availability vector $\mathbf{m}^{(s)}$, so that tokens corresponding to missing modalities (i.e., those generated purely from contextual null embeddings) neither attend to nor are attended by genuine modalities. Since the fusion module consumes the reliability-weighted embeddings $\hat{z}_i^{M(s)}$, the resulting cross-modal attention effectively treats $\alpha^{M(s)}$ as modality-specific confidence coefficients: modalities that are down-weighted by $\mathcal{G}$ participate only weakly in the attention computation and cannot dominate the fused representation. In this way, the combination of gating and masked multi-way fusion implements an implicit error-correction mechanism that attenuates the impact of noisy or unreliable modalities on $e_i^{(s)}$.
5. **Projection heads (P_1, P_2):** Two MLP-based projection heads map embeddings to specific latent spaces used for auxiliary loss calculations and subsequent prototype generation. Specifically, $\mathbf{r}_i^{(s)} = P_1(e_i^{(s)})$ projects the final fused embedding, and $\mathbf{z}_i^{M(s)} = P_2(\hat{z}_i^{M(s)})$ projects the individual pre-gating modality embeddings. These projected representations serve as direct inputs to the loss functions and as features for local prototype updates.
6. **Classifier (C):** A final classification head maps the fused embedding $e_i^{(s)}$ to class logits for computing the task loss.

Together, these architectural features provide a robust solution for handling missing modalities. The combination of contextual imputation ($\mathcal{N}_M$), adaptive weighting ($\mathcal{G}$), and masked fusion (F) enables the model to generate meaningful representations even from incomplete data. The outputs of this architecture, specifically the projected embeddings $\mathbf{r}_i^{(s)}$ and $\{\mathbf{z}_i^{M(s)}\}_M$, along with the final class logits, serve as the direct inputs for the multi-objective loss function, which is detailed in Section 4.4.

4.3. Prototype management

The prototype exchange, aggregation, and update steps of the protocol (Steps 1, 2, 4, and 5) are governed by the management of class prototypes. These prototypes are the core data structures for knowledge exchange in MODALIS. Clients manage their local prototypes $\mathbf{P}_i$ and compute aggregated prototypes $\bar{\mathbf{P}}_i$ based on information from neighbors. Each prototype, denoted $\mathbf{p}_i^j(*, k) \in \mathbb{R}^d$, is an L2-normalized vector representing class k for a specific type $*$. The type $*$ can refer to an individual modality M or the fused representation.

- **Prototype aggregation:** At the beginning of each communication round, a client computes its aggregated prototype library. The aggregated prototype $\bar{\mathbf{p}}_i^j(*, k)$ is formed via the count-weighted average specified in Eq. (2). This equation combines the client's local prototype from the previous round ($\mathbf{p}_i^{j-1}(*, k)$) with the most recently received valid prototypes from its neighbors, where $\hat{t}_j$ in the equation denotes the originating round of the neighbor's prototype. This formulation is inherently asynchronous, as $\hat{t}_j$ may be older than $t - 1$ for neighbors that are slow to communicate, ensuring the protocol remains robust to network delays. The aggregation is weighted by the number of samples, $c_i^j(*, k)$, that contributed to each prototype, giving more influence to prototypes built from more data:

$$\bar{\mathbf{p}}_i^j(*, k) = \frac{c_i^{j-1}(*, k)\mathbf{p}_i^{j-1}(*, k) + \sum_{j \in \mathcal{N}_i} c_j^{\hat{t}_j}(*, k)\mathbf{p}_j^{\hat{t}_j}(*, k)}{c_i^{j-1}(*, k) + \sum_{j \in \mathcal{N}_i} c_j^{\hat{t}_j}(*, k)} \quad (2)$$

This mechanism is crucial for mitigating non-IID class distributions, as it exposes clients to representations of classes that may be locally scarce. The weights $c_i^{j-1}(*, k)$ and $c_j^{\hat{t}_j}(*, k)$ encode how many training samples contributed to each modality and class-specific prototype, so prototypes from scarce or noisy data have less influence than those supported by many samples. Thus, Eq. (2) performs data-driven, modality-aware reliability weighting, and the aggregated prototypes used as targets in the federated alignment and contrastive losses serve as anchors that pull noisy or underrepresented embeddings toward more reliable regions of the feature space. The resulting vector is L2-normalized.

- **Local prototype update:** After local training, a client updates its local prototypes. It first computes the mean feature embedding $\bar{\mathbf{z}}_k^*$ for each class k and type $*$ using genuine samples observed in the current epoch. The prototype is then updated via an adaptive Exponential Moving Average (EMA):

$$\mathbf{p}_i^j(*, k) \leftarrow \text{L2Norm}((1 - m_t) \cdot \text{L2Norm}(\mathbf{p}_i^{j-1}(*, k)) + m_t \cdot \text{L2Norm}(\bar{\mathbf{z}}_k^*)) \quad (3)$$

The adaptive momentum m_t allows for both stability and rapid adaptation to evolving feature distributions.

4.4. Multi-objective loss function

A composite, multi-objective loss function drives local model updates during training. This function, shown in Eq. (4), strategically leverages the aggregated prototypes $\bar{\mathbf{p}}_i^j(*, k)$ to balance multiple synergistic objectives and guide the model towards a robust, generalized solution.

$$\begin{aligned} L_{\text{total}} = & w_{\text{task}} L_{\text{task}} + w_{\text{FAL}} L_{\text{FAL}} + w_{\text{DCL}} L_{\text{DCL}} \\ & + w_{\text{MCL}} L_{\text{MCL}} \\ & + w_{\text{PR}} L_{\text{PR}} \end{aligned} \quad (4)$$

where w are scalar balancing hyperparameters. The individual components, computed over a mini-batch B , are defined as:

- **Task loss (L_{task}):** The primary objective loss for the downstream task is often implemented using a robust loss function like Focal Loss [27] applied to the classifier outputs $\mathbf{z}_{\text{cls}}^{(s)}$ and the ground truth labels

$y_i^{(s)}$ from the current mini-batch. The use of Focal Loss helps address the class imbalance that is often present in individual client datasets.

$$\mathcal{L}_{\text{task}} = \frac{1}{|B|} \sum_{s \in B} \mathcal{L}_{\text{cls}}(\mathbf{z}_{\text{cls}}^{(s)}, y_i^{(s)}) \quad (5)$$

- **Federated alignment loss ($\mathcal{L}_{\text{FAL}}$):** Aligns the projected fused embedding $\mathbf{r}_i^{(s)}$ with its corresponding aggregated fusion prototype, thereby encouraging neighborhood consensus on fused representations:

$$\mathcal{L}_{\text{FAL}} = \frac{1}{|B|} \sum_{s \in B} \left(\left\| \mathbf{r}_i^{(s)} - \text{StopGrad}(\bar{\mathbf{p}}_i^{\text{fusion}}, y_s) \right\|_2^2 \right) \quad (6)$$

The StopGrad operation prevents gradients from backpropagating into the aggregated prototypes, treating them as fixed targets for alignment during the current local update step.

- **Discriminative contrastive loss ($\mathcal{L}_{\text{DCL}}$):** Enhances intra-class compactness and inter-class separability for modality-specific embeddings using a margin-based contrastive loss against aggregated modality prototypes. Inspired by margin-based classification losses such as ArcFace [28], it can be formulated using a cross-entropy (CE) loss over scaled cosine similarity scores, where an angular margin μ is added to the angle corresponding to the target class prototype. For a genuine modality M in sample s :

$$\mathcal{L}_{\text{DCL}}^{(s,M)} = \text{CE} \left(\text{softmax} \left(\frac{1}{\tau_{\text{DCL}}} \left[s_{\text{DCL}} \cos(\theta_k + \mu \delta_{k,y_s}) \right]_{k=1}^K, y_s \right) \right) \quad (7)$$

where θ_k is the angle between $\mathbf{z}_i^{M,(s)}$ and $\bar{\mathbf{p}}_i^{\text{fusion}}(M, k)$, μ is an angular margin, and δ_{k,y_s} is the Kronecker delta. The total loss is averaged over all genuine modalities in the batch.

- **Modality coherence loss ($\mathcal{L}_{\text{MCL}}$):** Promotes semantic consistency between representations from different modalities of the same sample by maximizing their cosine similarity:

$$\mathcal{L}_{\text{MCL}} = \frac{1}{|B|} \sum_{s \in B} \sum_{\substack{M_a, M_b \in \mathcal{M}, M_a \neq M_b \\ m_s^{M_a}=1, m_s^{M_b}=1}} \left(1 - \frac{\mathbf{z}_i^{M_a,(s)} \cdot \mathbf{z}_i^{M_b,(s)}}{\left\| \mathbf{z}_i^{M_a,(s)} \right\| \left\| \mathbf{z}_i^{M_b,(s)} \right\|} \right) \quad (8)$$

In this sense, it encourages the model to learn shared semantic information across different views of the same underlying concept.

- **Prototype regularization loss ($\mathcal{L}_{\text{PR}}$):** Gently regularizes local prototypes to stay close to the aggregated ones, promoting network-wide stability:

$$\mathcal{L}_{\text{PR}} = \sum_{s,k} \left\| \mathbf{p}_i^{t-1}(s, k) - \text{StopGrad}(\bar{\mathbf{p}}_i^{\text{fusion}}(s, k)) \right\|_2^2 \quad (9)$$

In summary, the architectural components in Section 4.2 and the losses in Eq. (4) implement an implicit modality-specific error correction scheme. The gating module $\mathcal{G}$ learns reliability weights $\alpha^{M,(s)}$ that downweight noisy modalities, and the masked multiway transformer $\mathcal{F}$ prevents missing or imputed inputs from dominating the fused representation. The prototype-based losses $\mathcal{L}_{\text{FAL}}$ and $\mathcal{L}_{\text{DCL}}$ align modality embeddings with count-weighted aggregated prototypes dominated by well-supported observations. The cross-modality coherence term $\mathcal{L}_{\text{MCL}}$ enforces consistent semantics across modalities, while $\mathcal{L}_{\text{PR}}$ pulls local prototypes toward their neighborhood consensus. As a result, modalities are not fused uniformly but are adaptively weighted according to both learned reliability and the amount of supporting data for each modality and class.

5. Theoretical guarantees

This section presents a theoretical analysis of the convergence properties and communication efficiency of MODALIS, providing foundational guarantees for its behavior in decentralized environments.

5.1. Convergence analysis

The convergence analysis of MODALIS considers a setting with N clients, each optimizing a local objective function $F_i(\theta_i; \bar{\mathbf{P}}_i)$ that depends on its local model parameters θ_i and the aggregated prototypes $\bar{\mathbf{P}}_i$ available at round t . The global objective implicitly minimizes the average of these local objectives across all clients. Due to the decentralized nature, the use of peer-aggregated prototypes, and the non-convexity of deep learning objectives, the analysis typically aims to show that the algorithm converges to a stationary point or a neighborhood thereof.

The analysis relies on the following standard assumptions:

Assumption 1 (L-smoothness). Each local objective function $F_i(\theta; \bar{\mathbf{P}})$ is L -smooth with respect to its model parameters θ , for any fixed set of aggregated prototypes $\bar{\mathbf{P}}$. That is, for all θ, θ' , there exists a constant $L > 0$ such that $\left\| \nabla_{\theta} F_i(\theta; \bar{\mathbf{P}}) - \nabla_{\theta} F_i(\theta'; \bar{\mathbf{P}}) \right\| \leq L \|\theta - \theta'\|$.

Assumption 2 (Unbiased stochastic gradients with bounded variance). The stochastic gradient $\hat{\nabla}_{\theta} F_i(\theta_i; \bar{\mathbf{P}}_i)$ computed by client i on a mini-batch of size B at round t is an unbiased estimate of the true local gradient $\nabla_{\theta} F_i(\theta_i; \bar{\mathbf{P}}_i)$, i.e., $\mathbb{E}[\hat{\nabla}_{\theta} F_i(\theta_i; \bar{\mathbf{P}}_i) | \theta_i, \bar{\mathbf{P}}_i] = \nabla_{\theta} F_i(\theta_i; \bar{\mathbf{P}}_i)$. Furthermore, the variance of the stochastic gradient is bounded: $\mathbb{E} \left[\left\| \hat{\nabla}_{\theta} F_i(\theta_i; \bar{\mathbf{P}}_i) - \nabla_{\theta} F_i(\theta_i; \bar{\mathbf{P}}_i) \right\|^2 | \theta_i, \bar{\mathbf{P}}_i \right] \leq \sigma^2/B$ for some constant $\sigma^2 \geq 0$.

Assumption 3 (Bounded gradients). The expected squared norm of the stochastic gradients is uniformly bounded: $\mathbb{E} \left[\left\| \hat{\nabla}_{\theta} F_i(\theta_i; \bar{\mathbf{P}}_i) \right\|^2 \right] \leq G^2$ for all clients i and rounds t , for some constant $G^2 > 0$. This implies $\mathbb{E} \left[\left\| \nabla_{\theta} F_i(\theta_i; \bar{\mathbf{P}}_i) \right\|^2 \right] \leq G^2$.

Assumption 4 (Network connectivity and mixing). The communication graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ is connected. The prototype aggregation process, which averages neighbors' prototypes, is modeled as a stochastic mixing process. Let W_t be a time-varying, row-stochastic matrix representing the communication at round t , where $W_{t,ij} > 0$ only if client i successfully receives a prototype message from client j during that round. Despite asynchronous, potentially delayed communication (i.e., bounded delay), the process is assumed to mix information effectively over time. The rate of convergence towards consensus is related to the spectral properties of the expected mixing matrix $\mathbb{E}[W_t]$, characterized by a parameter $\rho < 1$. Intuitively, better-connected networks (those with larger spectral gaps and smaller diameters) admit smaller values of ρ , leading to faster dissemination of information. In many common random graph models with bounded average degree (e.g., small-world or expander-like graphs), the spectral gap remains bounded away from zero as N grows, so ρ does not approach 1 and prototype information continues to mix efficiently even in large networks. In contrast, very sparse or elongated graphs such as rings or chains have spectral gaps that shrink quickly with N , which corresponds to ρ approaching 1 and a slower decay of the consensus error term in Theorem 1. This dependence on connectivity highlights that the convergence speed of MODALIS is governed by the structural properties of the chosen topology and its effective diameter, even though each round involves only local, neighbor-to-neighbor communication. From a modeling perspective, moderate topology dynamics and temporary link failures can therefore be represented by the time-varying sequence W_t , provided that the union of active links remains connected over sufficiently long windows.

Assumption 5 (Bounded prototype influence and drift). The gradient $\nabla_{\theta} F_i(\theta_i; \bar{\mathbf{P}}_i)$ exhibits bounded sensitivity to changes in the aggregated prototypes $\bar{\mathbf{P}}_i$. Specifically, it is assumed that for any θ_i , $\left\| \nabla_{\theta} F_i(\theta_i; \bar{\mathbf{P}}_i) - \nabla_{\theta} F_i(\theta_i; \bar{\mathbf{P}}_i') \right\| \leq L_p \|\bar{\mathbf{P}}_i - \bar{\mathbf{P}}_i'\|$ for some Lipschitz constant $L_p > 0$. Furthermore, the expected squared change (drift) in local prototypes per round is bounded, i.e., $\mathbb{E} \left[\left\| \mathbf{p}_i^{t-1}(s, k) - \mathbf{p}_i^{t-1}(s, k) \right\|^2 \right] \leq \delta_p^2$ for all

$i, t, *, k$. The discrepancy between a client's aggregated prototypes $\bar{\mathbf{P}}_i^t$ and a hypothetical global average prototype $\mathbf{P}_{\text{global}}^*$ is bounded, and this bound $E_{\text{proto, disp}}$ is influenced by network mixing properties (quantified by ρ) and the degree of data heterogeneity H across clients.

Theorem 1 (Convergence to a stationary neighborhood). *Under Assumptions 1–5, for MODALIS (Algorithm 1) with $T = 1$ local epoch per round and a constant learning rate η chosen sufficiently small (e.g., $\eta L \leq 1/2$), the average expected squared gradient norm across all clients and rounds converges to a neighborhood of zero:*

$$\frac{1}{R} \sum_{t=0}^{R-1} \frac{1}{N} \sum_{i=1}^N \mathbb{E} \left[\left\| \nabla_{\theta_i} F_i(\theta_i^t; \bar{\mathbf{P}}_i^t) \right\|^2 \right] \leq \mathcal{O} \left(\frac{F_0 - F^*}{\eta R} + \eta G^2 + \frac{\sigma^2}{B} + \epsilon(\eta, \rho, \delta_p, H, L_p, E_{\text{proto, disp}}) \right) \quad (10)$$

where R is the total number of communication rounds, F_0 is the initial global objective value, F^* is a conceptual minimum, B is the mini-batch size, and $\epsilon(\cdot)$ is a residual error term. This term ϵ captures the aggregate impact of imperfect prototype consensus due to decentralized aggregation (related to ρ and heterogeneity H via $E_{\text{proto, disp}}$), the evolution of the local objectives due to changing prototypes (related to δ_p and L_p), and stochastic noise. As $R \rightarrow \infty$, for sufficiently small η , the algorithm is guaranteed to reach a region where the average expected squared gradient norm is bounded by $\mathcal{O}(\eta G^2 + \sigma^2/B + \epsilon(\cdot))$.

Proof sketch. The proof adapts standard convergence analysis techniques for decentralized non-convex optimization [29,30] and FL with evolving objectives or non-identical data [7,25].

1. Start with the descent lemma for L -smooth functions (Assumption 1) applied to the local objective $F_i(\theta_i^t; \bar{\mathbf{P}}_i^t)$ for client i at round t : $\mathbb{E}[F_i(\theta_i^{t+1}; \bar{\mathbf{P}}_i^t)] \leq F_i(\theta_i^t; \bar{\mathbf{P}}_i^t) - \eta \mathbb{E}[\langle \nabla_{\theta_i} F_i(\theta_i^t; \bar{\mathbf{P}}_i^t), \hat{\nabla}_{\theta_i} F_i(\theta_i^t; \bar{\mathbf{P}}_i^t) \rangle] + \frac{L\eta^2}{2} \mathbb{E}[\left\| \hat{\nabla}_{\theta_i} F_i(\theta_i^t; \bar{\mathbf{P}}_i^t) \right\|^2]$.
2. Using Assumptions 2 and 3, and for $\eta L \leq 1/2$, this simplifies to: $\mathbb{E}[F_i(\theta_i^{t+1}; \bar{\mathbf{P}}_i^t)] \leq F_i(\theta_i^t; \bar{\mathbf{P}}_i^t) - \frac{\eta}{2} \mathbb{E}[\left\| \nabla_{\theta_i} F_i(\theta_i^t; \bar{\mathbf{P}}_i^t) \right\|^2] + \frac{\eta\sigma^2}{2B} + \frac{L\eta^2 G^2}{2}$.
3. Account for the change in the objective function due to prototype evolution: $F_i(\theta_i^{t+1}; \bar{\mathbf{P}}_i^{t+1}) - F_i(\theta_i^{t+1}; \bar{\mathbf{P}}_i^t)$. This difference can be bounded using Assumptions 5 and 4, which introduces error terms dependent on L_p, δ_p, ρ , and H .
4. Average the inequality over all N clients. Define a global potential function $\Phi^t = \frac{1}{N} \sum_{i=1}^N F_i(\theta_i^t; \bar{\mathbf{P}}_i^t)$. Sum the averaged inequalities over communication rounds $t = 0, \dots, R-1$.
5. Apply a telescoping sum argument to $\mathbb{E}[\Phi^{t+1}] - \mathbb{E}[\Phi^t]$. Rearrange terms to isolate the average squared gradient norm $\frac{1}{RN} \sum_{t=0}^{R-1} \sum_{i=1}^N \mathbb{E}[\left\| \nabla_{\theta_i} F_i(\theta_i^t; \bar{\mathbf{P}}_i^t) \right\|^2]$.
6. The remaining terms constitute the right-hand side of the theorem. The term $\epsilon(\cdot)$ aggregates errors from stochastic noise, imperfect prototype consensus, and the drift in local objectives. The use of $T = 1$ local epoch simplifies the analysis by avoiding client drift complexities [25].

The theorem indicates that MODALIS converges to a neighborhood of a stationary point, where the size of the neighborhood is influenced by the learning rate, stochastic gradient variance, mini-batch size, and the compounded error term ϵ . $\square$

5.2. Communication efficiency

Lemma 1 (Prototype communication cost reduction). *Exchanging class prototypes in MODALIS yields a significant reduction in communication overhead per round compared to DFL methods that exchange full model parameters. Let $|\theta|$ denote the total number of parameters in the client model architecture, d be the dimension of the prototype embedding vectors, K be*

the total number of distinct classes, and $|\mathcal{M}_{\text{proto}}|$ be the number of distinct prototype types exchanged per class. Assuming a numerical precision p (e.g., 32 bits), the approximate per-round communication cost for a client to send its prototypes to one neighbor is $C_{\text{proto}} = K \cdot |\mathcal{M}_{\text{proto}}| \cdot d \cdot p$ bits. In contrast, standard DFL scenarios that exchange full model parameters incur a cost of $C_{\text{model}} = |\theta| \cdot p$ bits. Given that for typical deep learning models, $K \cdot |\mathcal{M}_{\text{proto}}| \cdot d \ll |\theta|$, MODALIS achieves substantial communication savings.

Proof. In each round, client i transmits its prototype library $\mathbf{P}_i^t$ to its neighbors. This library contains $K \times |\mathcal{M}_{\text{proto}}|$ vectors, each with d values. The total data transmitted to one neighbor is $K \cdot |\mathcal{M}_{\text{proto}}| \cdot d \cdot p$ bits. Conversely, DFL methods exchanging full models transmit the entire parameter vector θ_i of size $|\theta|$, for a cost of $C_{\text{model}} = |\theta| \cdot p$ bits. For a practical example, if $K = 100$, $|\mathcal{M}_{\text{proto}}| = 3$, and $d = 256$, the number of values is $100 \times 3 \times 256 = 76,800$. If the model has $|\theta| = 10^7$ parameters, the cost ratio is $C_{\text{model}}/C_{\text{proto}} = 10^7/76,800 \approx 130$. This reduction of one to three orders of magnitude directly translates to reduced bandwidth requirements and faster training time, especially when communication is the bottleneck. $\square$

5.3. Privacy considerations

Sharing learned class prototypes, rather than raw model parameters or gradients, intuitively offers potential advantages regarding data privacy. Prototypes are abstract, aggregated representations summarizing the characteristic features of entire classes, learned from multiple local data samples. They do not directly expose information about individual data points, as gradients (which are directly tied to specific mini-batches) or model parameters might. Compared to sharing gradients, which is vulnerable to sophisticated inference attacks such as Deep Leakage from Gradients (DLG) [31], prototype sharing is generally considered to present a reduced attack surface [7]. The aggregation process inherent in forming prototypes further obfuscates individual contributions.

However, MODALIS in its base form does not provide formal privacy guarantees, such as those offered by Differential Privacy (DP) [32]. In the standard DP framework, a randomized mechanism $\mathcal{M}$ that releases messages (e.g., prototype libraries) is said to satisfy (ϵ, δ) -DP if, for any pair of neighboring datasets D and D' differing in a single sample and for any measurable set S in the output space, it holds that $\Pr[\mathcal{M}(D) \in S] \leq e^\epsilon \Pr[\mathcal{M}(D') \in S] + \delta$. The current design neither injects calibrated noise into the shared prototypes nor enforces this constraint, and therefore it does not offer any formal (ϵ, δ) -DP guarantee. Potential privacy risks may still arise; for example, through meticulous analysis of sequences of prototypes on clients where only a few samples represent a given class. In such cases, the corresponding class prototype may implicitly encode characteristics closer to those of individual examples. As a consequence, standard membership inference or related privacy attacks remain, in principle, applicable in low-support regimes, even though the use of aggregated prototypes is generally expected to be more resilient than raw gradient sharing in typical settings.

6. Experimentation

This section details the comprehensive empirical evaluation of MODALIS. The experiments were meticulously designed to: (i) assess its overall classification performance against relevant FL baselines across a diverse suite of multimodal datasets; (ii) evaluate its robustness and adaptability under systematically varying degrees of data heterogeneity, specifically non-IID class distributions and different levels of missing modalities; (iii) quantify its communication efficiency and estimate total training time; and (iv) analyze the contribution of its key architectural and algorithmic components through rigorous ablation studies.

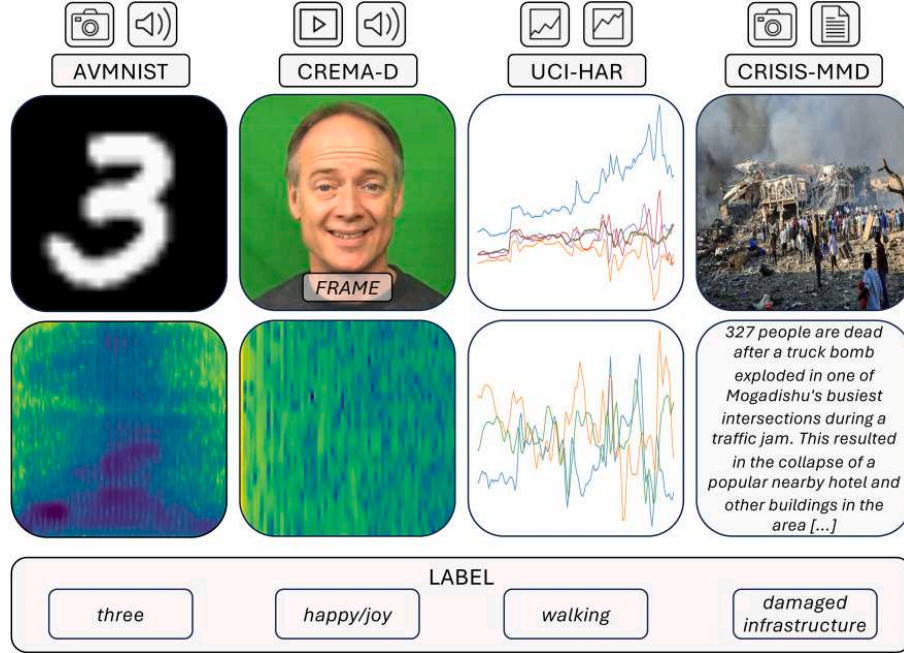


Fig. 3. Example inputs and labels from four multimodal datasets used in this work.

6.1. Experimental setup

- **Datasets:** Four publicly available multimodal datasets were selected to represent a variety of tasks and modality combinations: **AVMNIST** [33] (audio-visual spoken digits), **CREMA-D** [34] (audio-visual emotional speech), **UCI-HAR** [35] (accelerometer and gyroscope data for human activity recognition), and **CRISIS-MMD** [36] (image and text from social media for crisis event classification). These datasets span a spectrum of complexities, from simple digit recognition to nuanced emotional and situational analysis (see Fig. 3).
- **Model architectures:** Modality-specific encoders ($\mathcal{E}_M$) were chosen based on common practices for each data type. The fusion module ($\mathcal{F}$) and other shared components (projection heads P_1, P_2 , classifier C) were kept consistent where possible to ensure fair comparisons.
 - AVMNIST: A simple Convolutional Neural Network (CNN, similar to LeNet-5) for images, and a 3-layer 2D CNN for spectrograms.
 - CREMA-D: A ResNet-18 model [37] processing averaged frames for the video modality, and a 2-layer Bi-directional Long Short-Term Memory network (BiLSTM) operating on MFCC features for the audio modality.
 - UCI-HAR: Two parallel 4-layer 1-D CNNs, each processing the respective 3-axis sensor time series data.
 - CRISIS-MMD: A ResNet-18 model for images, and a Bidirectional Encoder Representations from Transformers (BERT)-base model [38] for text.
- **Heterogeneity simulation:** To mimic challenging real-world federated scenarios, data heterogeneity was introduced artificially across a simulated network of $N = 20$ clients (see Fig. 4 for an illustration of random clients). This heterogeneity has two primary dimensions:
 - **Non-IID class distributions:** The data was partitioned among clients using a Dirichlet distribution $Dir(\beta)$, a standard technique for simulating label skew [39]. Experiments were conducted with

$\beta \in \{1.0, 0.5, 0.1\}$, where a smaller β corresponds to a higher degree of skew.

- **Missing modalities:** The proportion of missing modalities was controlled by a parameter $p_m \in \{0.0, 0.25, 0.5\}$. For a given $p_m > 0$, a fraction p_m of clients received data exclusively from one modality, another fraction p_m received data exclusively from the other, and the remaining $1 - 2p_m$ fraction of clients received complete multimodal data. This setup creates a challenging environment with both fully unimodal and multimodal clients.

- **Baselines and federation topology:** MODALIS was benchmarked against a suite of representative methods: **Centralized** (a theoretical upper bound trained on all data), **FedAvg** [1], **FedProto** [7], **FedInMM** [21], **CreamFL** [22], **PmcmFL** [17], **FedMAC** [23], and **MFCPL** [24]. For a fair comparison with MODALIS, all federated methods were simulated with $N = 20$ participating clients connected via a static Erdős-Rényi (ER) random graph with edge probability $p_{ER} = 0.9$, which provides a realistic but controlled decentralized connectivity pattern for the main reported results. In addition, Section 6.5 includes a dedicated topology-sensitivity analysis in which NEBULA is used to instantiate multiple decentralized graphs to assess the robustness of MODALIS to changes in network structure.

6.2. Implementation details

Training configuration: The framework and all baselines were implemented in PyTorch 2.0 and integrated into NEBULA, an open-source research framework for DFL [40]. NEBULA was selected because it provides a topology-agnostic and model-agnostic environment in which different communication graphs, numbers of clients, and algorithmic variants can be instantiated in a unified and reproducible way. In particular, NEBULA exposes a modular communication layer that allows the same

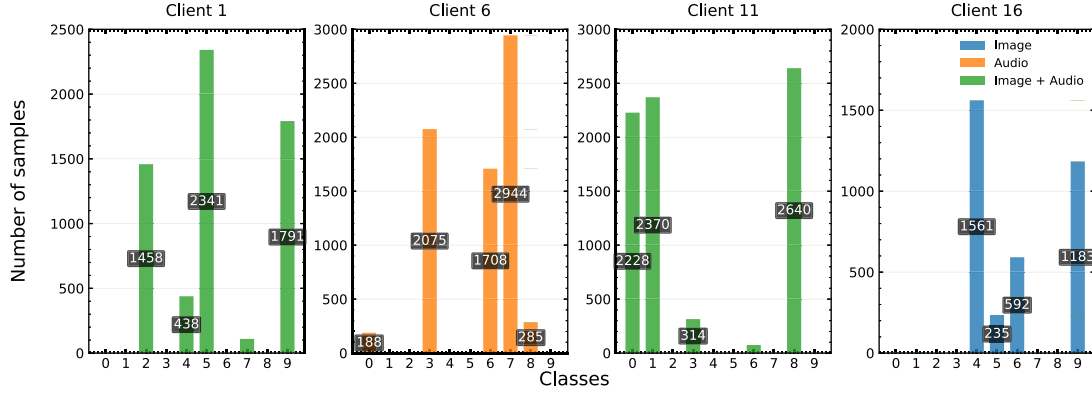


Fig. 4. Four random client data distributions for AVMNIST under specified heterogeneity parameters ($\beta = 0.1$ for class skew, $p_m = 0.25$ for missing modality proportion).

codebase to be executed over a variety of decentralized topologies and custom prototype-exchange protocols, and it supports container-based deployments for running algorithms in realistic multi-process settings. In this work, NEBULA is primarily used in its simulation mode to isolate the algorithmic behavior of MODALIS under controlled heterogeneity and topology configurations, while leaving large-scale containerized deployments with explicit network latencies, client churn, and node mobility to future work. All models were trained using the AdamW optimizer with a learning rate of $\eta = 2 \times 10^{-4}$ and a batch size of $B = 64$. Experiments were conducted for a total of $R = 200$ communication rounds with a common embedding dimension of $d = 128$, being a practical trade-off between representation capacity and communication cost. The framework supports larger embedding dimensions and provides slightly higher accuracy at the expense of proportionally larger prototype messages (see Section 6.5).

Hyperparameter selection for MODALIS: The hyperparameters for MODALIS, particularly the loss weights in Eq. (4), were determined via a grid search on a held-out validation set using the AVMNIST dataset under moderate heterogeneity ($\beta = 0.5$, $p_m = 0.25$). The chosen values, used consistently across all experiments, reflect the relative importance of each objective. The final weights are: $w_{\text{task}} = 1.0$, $w_{\text{FAL}} = 0.5$, $w_{\text{DCL}} = 0.2$, $w_{\text{MCL}} = 0.1$, and $w_{\text{PR}} = 0.01$. This configuration prioritizes the primary task loss, assigns significant weight to the federated alignment loss ($\mathcal{L}_{\text{FAL}}$) to achieve consensus, and uses the remaining terms as auxiliary regularizers to structure the embedding space without overpowering the main objectives. For the $\mathcal{L}_{\text{DCL}}$, the temperature was set to $\tau_{\text{DCL}} = 0.1$, the angular margin to $\mu = 0.3$ radians, and the scale to $s_{\text{DCL}} = 30$, following common practices in representation learning [28]. The adaptive EMA momentum parameters were bounded by $m_{\min} = 0.5$ and $m_{\max} = 0.99$, with the update logic detailed in Algorithm 1.

Evaluation protocol: To ensure robust conclusions, the reported results are the average and standard deviation over three independent runs with different random seeds. The experimental testbed consisted of a compute cluster node equipped with two NVIDIA 3060 GPUs and 128 GB of RAM. Timing estimations assume typical network latencies proportional to data size for the communication phases.

6.3. Performance and convergence comparison

The performance and robustness of MODALIS were evaluated against the baseline suite under two distinct levels of data heterogeneity. First, a moderate-stress scenario with mild label skew ($\beta = 0.5$) was analyzed, followed by a high-stress scenario with severe label skew ($\beta = 0.1$), both of which were combined with increasing proportions of missing

modalities (p_m). Performance was primarily assessed using the Average Macro-F1 Score, a metric chosen for its robustness to the class imbalance created by the non-IID simulation. To provide a holistic evaluation, Communication Cost (in Megabytes per client per round) and Estimated Total Training Time were also reported to quantify efficiency.

6.3.1. Performance under moderate label skew ($\beta = 0.5$)

This setting provides insight into the methods' robustness to missing modalities when label skew is present but less extreme. Table 3 presents the F1 scores and relative drops for this scenario, while Fig. 5 provides a chart visualization of the model performance using $\beta = 0.5$. The results show that MODALIS consistently achieves top-tier performance and remarkable robustness. It either matches or outperforms all federated baselines on AVMNIST, CREMA-D, and CRISIS-MMD across all levels of missing modalities, and remains highly competitive on UCI-HAR, where the MFCPL baseline attains a slightly higher Macro-F1 in several configurations. A key finding is its resilience to performance degradation. For instance, on the UCI-HAR dataset, MODALIS decreases from 92.2% to 84.5% when moving from complete data ($p_m=0.0$) to maximal missingness ($p_m=0.5$), which corresponds to a relative loss of roughly 7.7%. Most competing methods lose between 9% and 13% in the same setting. From a dataset-specific perspective, MFCPL emerges as the strongest baseline on UCI-HAR, surpassing MODALIS by 0.5–0.6% across $p_m=0.25-0.5$. This behavior is consistent with its design, which tightly couples class-wise prototypes across modalities and is particularly effective when accelerometer and gyroscope streams are highly correlated.

6.3.2. Performance under high label skew ($\beta = 0.1$)

The high-skew setting represents a significant stress test for any FL algorithm, as it challenges the ability to form coherent global knowledge from highly biased local data. The results in Table 4 show that MODALIS becomes even more critical in this scenario. It achieves the best overall performance across datasets and missing-modality levels, with particularly large margins on AVMNIST and CRISIS-MMD. At the same time, MFCPL is closely matched on UCI-HAR, attaining slightly higher Macro-F1 scores (within 0.5–0.6 points) across all p_m values. However, the relative performance drop of MFCPL with increasing p_m remains slightly larger than MODALIS, indicating that the proposed neighborhood-aware prototype regularization is more robust to extreme information loss when averaging across heterogeneous clients. Its advantage over the standard FedAvg was particularly pronounced, with FedAvg collapsing as heterogeneity increased. For example, at $p_m = 0.5$ on AVMNIST, MODALIS achieved a relative F1 score increase of +20.0% over FedAvg. Fig. 6 illustrates the convergence behavior in the most extreme case

Table 3
Performance under moderate label skew ($\beta = 0.5$).

Method	AVMNIST				CREMA-D				UCI-HAR				CRISIS-MMD			
	$p_m=0.0$	$p_m=0.25$	$p_m=0.5$	$(0 \rightarrow 0.5)^\dagger$	$p_m=0.0$	$p_m=0.25$	$p_m=0.5$	$(0 \rightarrow 0.5)^\dagger$	$p_m=0.0$	$p_m=0.25$	$p_m=0.5$	$(0 \rightarrow 0.5)^\dagger$	$p_m=0.0$	$p_m=0.25$	$p_m=0.5$	$(0 \rightarrow 0.5)^\dagger$
Centralized	98.5	94.0	89.0	13.5%	76.0	69.0	62.1	13.9%	95.8	90.0	87.2	8.6%	81.0	74.5	67.9	13.7%
FedAvg	87.3	81.7	74.2	13.1%	59.3	52.7	45.3	14.0%	84.2	78.6	70.9	13.3%	64.1	57.5	50.1	14.0%
FedProto	91.2	86.4	79.3	11.9%	64.1	58.6	51.2	12.9%	88.1	83.4	76.5	11.6%	69.0	62.6	55.9	13.1%
FedInMM	92.4	88.7	83.1	9.3%	67.2	61.9	55.9	11.3%	90.3	86.7	81.2	9.1%	72.2	66.0	59.0	13.2%
CreamFL	91.8	87.6	81.2	10.6%	66.3	60.8	53.7	12.6%	89.4	84.9	78.3	11.1%	71.1	64.9	57.2	13.9%
PmemFL	93.2	89.5	84.0	9.2%	68.1	62.7	56.8	11.3%	91.3	87.6	82.1	9.2%	73.1	66.8	59.1	14.0%
FedMAC	93.0	89.0	83.8	9.2%	67.8	62.3	56.2	11.6%	91.0	87.2	81.7	9.3%	72.8	66.3	53.5	19.3%
MFCPL	93.5	89.8	84.7	8.8%	68.5	63.0	58.9	9.6%	93.1	88.7	83.9	9.2%	73.5	67.0	58.3	15.2%
MODALIS	94.1	90.1	86.3	7.8%	69.0	62.1	60.1	8.9%	92.2	88.2	84.5	7.7%	74.0	67.4	59.1	14.9%

[†] represents the performance decrease from $p_m = 0.0$ to $p_m = 0.5$. Centralized results are from a single run.

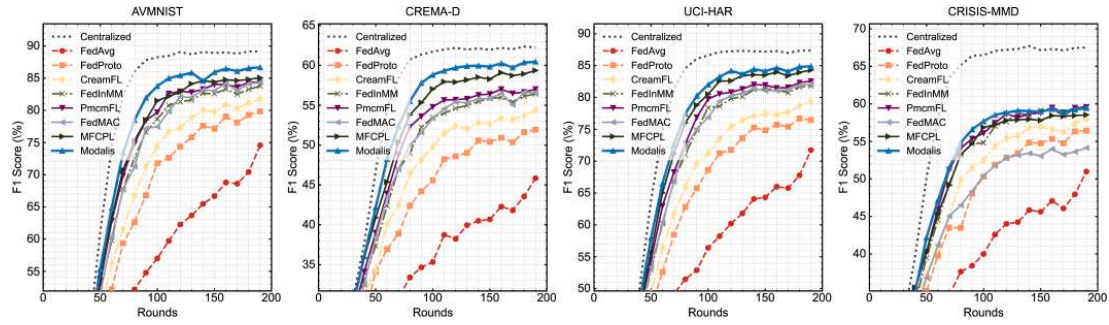


Fig. 5. Model performance (Avg. Macro-F1 Score %) under moderate Non-IID Skew ($\beta = 0.5$) and 50% missing modalities ($p_m = 0.5$).

Table 4
Performance under high label skew ($\beta = 0.1$).

Method	AVMNIST				CREMA-D				UCI-HAR				CRISIS-MMD			
	$p_m=0.0$	$p_m=0.25$	$p_m=0.5$	$(0 \rightarrow 0.5)^\dagger$	$p_m=0.0$	$p_m=0.25$	$p_m=0.5$	$(0 \rightarrow 0.5)^\dagger$	$p_m=0.0$	$p_m=0.25$	$p_m=0.5$	$(0 \rightarrow 0.5)^\dagger$	$p_m=0.0$	$p_m=0.25$	$p_m=0.5$	$(0 \rightarrow 0.5)^\dagger$
Centralized	97.6	92.8	87.9	9.7%	74.5	67.5	60.8	13.7%	94.7	88.5	81.8	12.9%	79.5	72.6	65.7	13.8%
FedAvg	86.5	79.2	69.5	17.0%	57.2	50.1	42.8	14.4%	81.5	74.3	65.7	15.8%	61.3	54.1	47.2	14.1%
FedProto	90.3	84.1	74.8	15.5%	62.1	56.3	48.7	13.4%	86.2	79.1	71.3	14.9%	66.2	59.3	53.8	12.4%
FedInMM	92.8	86.9	79.7	13.1%	64.8	58.4	52.6	12.2%	88.8	81.9	75.8	13.0%	68.7	61.7	58.1	10.6%
CreamFL	92.3	85.8	78.3	14.0%	64.1	57.2	51.1	13.0%	88.1	80.3	73.2	14.9%	68.1	60.7	56.7	11.4%
PmemFL	93.4	87.5	80.3	13.1%	65.3	58.9	54.1	11.2%	88.9	82.5	76.3	12.9%	67.3	60.1	54.2	13.1%
FedMAC	93.1	87.0	79.9	13.2%	65.9	58.6	53.5	12.4%	88.8	82.2	76.1	12.7%	67.2	59.2	55.7	11.5%
MFCPL	93.0	87.2	80.0	13.0%	65.0	58.7	53.8	11.2%	89.5	83.7	79.9	9.6%	68.5	61.5	57.8	10.7%
MODALIS	94.3	88.1	83.4	10.9%	66.2	59.5	55.1	11.1%	88.9	83.2	79.3	9.6%	70.2	61.7	60.0	10.2%

[†] represents the performance decrease from $p_m = 0.0$ to $p_m = 0.5$. Centralized results are from a single run.

($p_m = 0.5$), showing the learning dynamics under severe data fragmentation.

In this high-heterogeneity regime ($\beta = 0.1$ with p_m up to 0.5), the centralized upper bound already exhibits notable degradation, with Macro-F1 reductions between 9.7% and 13.8% when moving from $p_m = 0.0$ to $p_m = 0.5$ in Table 4. Under the same conditions, all federated methods incur additional losses compared to milder settings, and MODALIS is no exception. For AVMNIST, for example, MODALIS decreases from 94.3% at $p_m = 0.0$ to 83.4% at $p_m = 0.5$, corresponding to a relative drop of 10.9%, whereas FedAvg and FedProto degrade by 17.0% and 15.5%, respectively, over the same range. Similar behavior is observed on CREMA-D, UCI-HAR, and CRISIS-MMD, where MODALIS consistently shows the smallest relative performance loss among all compared methods. These results indicate that the prototype-based multimodal design substantially improves robustness to target heterogeneity, while not eliminating the inherent performance penalties associated with extreme information loss.

6.3.3. Cross-modal knowledge transfer on unimodal clients

To directly assess the ability of different frameworks to transfer useful knowledge across modalities, the performance of client groups that possessed only a single modality was analyzed in the most challenging setting ($p_m = 0.5, \beta = 0.1$). Table 5 reports the F1 scores for these spe-

cific unimodal clients. The results demonstrate the strong capacity of MODALIS for cross-modal knowledge transfer. On AVMNIST and CRISIS-MMD, unimodal clients using MODALIS achieve the highest F1 scores across all methods, and on CREMA-D, they are essentially tied with the best-performing baseline. For instance, on the AVMNIST dataset, MODALIS image-only clients achieved an F1 score of 82.1%, representing a 19.9% relative gain over FedAvg and outperforming all other advanced baselines. On UCI-HAR, MFCPL slightly surpasses MODALIS for both accelerometer-only and gyroscope-only clients, reflecting the effectiveness of its class-wise prototype matching in this particular sensor configuration.

6.3.4. Cross-dataset transferability

While the primary experiments train and evaluate MODALIS on each dataset separately, it is also important to understand how well a configuration tuned in one environment transfers to other, dissimilar tasks. In the present setup, the datasets differ not only in distribution but also in label spaces and, in some cases, modality semantics (e.g., spoken digits versus social-media crisis posts). As a result, fully zero-shot deployment of a model trained on AVMNIST directly to CRISIS-MMD, without retraining any parameters, is not well-defined: the class sets, loss functions, and input domains are incompatible. Instead, a practically rele-

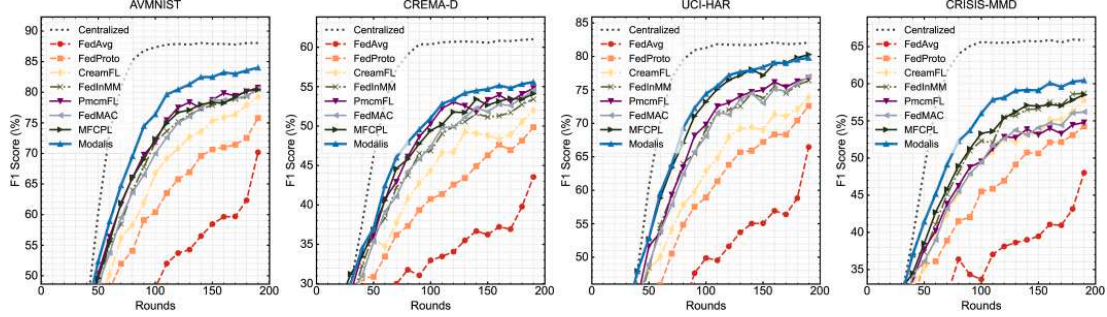


Fig. 6. Model performance (Avg. Macro-F1 Score %) under high Non-IID skew ($\beta = 0.1$) and 50% missing modalities ($p_m = 0.5$).

Table 5

Performance on client subgroups possessing only a single modality, evaluated under the high heterogeneity setting ($p_m = 0.5, \beta = 0.1$).

Method ^a	AVMNIST		CREMA-D		UCI-HAR		CRISIS-MMD	
	Image	Audio	Image	Audio	Accelerometer	Gyroscope	Text	Image
FedAvg	68.5±1.0	67.2±1.1	40.1±1.4	38.7±1.5	64.3±1.2	62.9±1.3	45.8±1.6	43.9±1.7
FedProto	73.1±0.8	71.5±0.9	46.2±1.1	44.9±1.2	69.8±0.9	68.3±1.0	51.9±1.3	50.0±1.4
FedInMM	78.3±0.7	76.8±0.8	50.2±1.0	48.8±1.1	74.3±0.8	73.0±0.9	56.3±1.1	54.5±1.2
CreamFL	76.5±0.9	74.9±1.0	48.1±1.2	46.7±1.3	71.6±1.0	70.2±1.1	53.6±1.4	51.7±1.5
PmcmFL	79.0±0.6	77.4±0.7	51.5±0.9	50.1±1.0	75.0±0.7	73.6±0.8	57.5±1.0	55.6±1.1
FedMAC	78.8±0.6	77.0±0.7	51.0±0.9	49.5±1.0	74.8±0.7	73.3±0.8	57.0±1.0	55.3±1.1
MFCPL	80.0±0.6	78.5±0.7	52.8±0.8	51.5±0.9	78.6±0.6	77.3±0.7	58.0±0.9	56.1±1.0
MODALIS	82.1±0.5	80.5±0.6	53.0±0.9	51.7±1.0	78.1±0.6	76.8±0.7	58.9±1.0	57.0±1.1

^a These clients use only their available modality for inference, but their models are trained within the respective FL/DFL frameworks.

vant notion of cross-dataset transferability is considered, in which the algorithmic configuration (hyperparameters and architectural choices) is tuned on a source dataset and then reused without modification when training on other target datasets.

To this end, two settings are compared for MODALIS, FedAvg, and the multimodal baseline MFCPL on CREMA-D, UCI-HAR, and CRISIS-MMD under moderate heterogeneity ($\beta = 0.5, p_m = 0.25$): (i) an oracle per-dataset tuning, where a small grid search is allowed on each target dataset; and (ii) an AVMNIST-tuned configuration, in which all hyperparameters are fixed to the values selected on AVMNIST and are not re-optimized on the target. The results, summarized in Table 6, show that MODALIS incurs only a modest performance degradation when transferring its AVMNIST-tuned configuration: on CREMA-D, UCI-HAR, and CRISIS-MMD, the Macro-F1 decreases by approximately 1.9, 1.2, and 1.9 points, respectively, while still attaining the best absolute scores across all methods. By contrast, FedAvg shows larger relative drops of 4.1% – 6.1% under the same protocol. MFCPL occupies an intermediate position: it clearly improves over FedAvg in both tuned and transferred configurations, but its transfer-induced degradation remains noticeably higher than that of MODALIS, with decreases of about 3.6, 2.0, and 3.5 points on CREMA-D, UCI-HAR, and CRISIS-MMD, respectively. These findings indicate that some dataset-specific tuning remains beneficial in practice, but MODALIS is the most robust to cross-dataset hyperparameter transfer. A single configuration tuned on AVMNIST yields near-oracle performance on other tasks, whereas both FedAvg and MFCPL are more sensitive to per-dataset tuning.

6.4. Resource cost and training time analysis

As established theoretically in Lemma 1, MODALIS achieves substantial communication efficiency by exchanging low-dimensional prototypes rather than full model parameters. This section complements that analysis with a quantitative study of the overall resource footprint, including computation time and node-level CPU, GPU, and RAM

utilization (see Table 7). The results show that a MODALIS client transmits approximately 0.11 MB per round, which is more than 40× smaller than communication-heavy methods such as FedAvg and PmcmFL (each sending around 4.75–4.80 MB) and is still substantially below advanced multimodal baselines such as FedInMM and FedMAC. This reduction in communication directly translates into faster communication-bound rounds in bandwidth-constrained environments.

The reported CPU and GPU usages primarily reflect the cost of batched tensor operations (forward and backward passes) in the modality encoders, multi-way transformer fusion, adaptive gating, projection heads, and loss computations, which are offloaded to the GPU, together with orchestration, PyTorch runtime overhead, data loading, and communication management handled by the CPU. RAM utilization captures the joint footprint of model parameters, intermediate activations, and small prototype buffers. As shown in Table 7, MODALIS exhibits GPU utilization of approximately 70% and RAM utilization of roughly 31% on the evaluation node, which is very close to multimodal FL methods with similarly rich architectures such as PmcmFL, FedMAC, FedInMM, and MFCPL. CPU utilization for MODALIS (approximately 44%) is likewise on par with these advanced baselines. In contrast, FedAvg and FedProto show somewhat lower GPU and RAM usage, but require significantly more communication per round. Overall, these results indicate that MODALIS targets moderately capable edge devices or edge servers, where the additional local computation relative to FedAvg or FedProto is on the same order as that of other state-of-the-art multimodal FL methods and is compensated by the substantial communication savings.

Finally, the drastically reduced communication payload, combined with moderate local resource usage, enables MODALIS to achieve competitive end-to-end convergence times. In the simulated, computation-intensive setting considered, MODALIS converges in approximately 4.5 hours, which is comparable to or faster than other advanced multimodal approaches such as FedInMM, FedMAC, PmcmFL, and MFCPL, while achieving the highest final Macro-F1 score of 83.4%. This behavior underscores that the prototype-centric design delivers a favorable trade-off

Table 6Cross-dataset hyperparameter transfer under moderate heterogeneity ($\beta = 0.5, p_m = 0.25$).

Method	Target dataset	Per-dataset tuned (%)	AVMNIST-tuned (%)	Δ (%)
FedAvg	CREMA-D	52.7 $\pm$ 0.6	49.7 $\pm$ 0.7	$\downarrow$ 5.7
	UCI-HAR	78.6 $\pm$ 0.5	75.4 $\pm$ 0.6	$\downarrow$ 4.1
	CRISIS-MMD	57.5 $\pm$ 0.7	54.0 $\pm$ 0.8	$\downarrow$ 6.1
MFCPL	CREMA-D	62.8 $\pm$ 0.4	59.2 $\pm$ 0.5	$\downarrow$ 3.6
	UCI-HAR	88.5 $\pm$ 0.3	86.5 $\pm$ 0.4	$\downarrow$ 2.0
	CRISIS-MMD	68.2 $\pm$ 0.4	64.7 $\pm$ 0.5	$\downarrow$ 3.5
MODALIS	CREMA-D	63.3 $\pm$ 0.4	62.1 $\pm$ 0.5	$\downarrow$ 1.9
	UCI-HAR	89.3 $\pm$ 0.3	88.2 $\pm$ 0.4	$\downarrow$ 1.2
	CRISIS-MMD	68.7 $\pm$ 0.4	67.4 $\pm$ 0.5	$\downarrow$ 1.9

Table 7Resource cost and training time using AVMNIST ($\beta = 0.1, p_m = 0.5$).

Method	Communication Cost (MB) ^a	Training Time (min) ^a	Total Time (hr)	CPU (%)	GPU (%)	RAM (%)	F1 score (%)
Centralized	N/A	N/A	$\approx$ 1.9	$\approx$ 52	$\approx$ 78	$\approx$ 34	87.9 $\pm$ 0.2
FedAvg	$\approx$ 4.75	$\approx$ 2.03	$\approx$ 6.1	$\approx$ 37	$\approx$ 54	$\approx$ 26	69.5 $\pm$ 0.9
FedProto	$\approx$ 0.03	$\approx$ 1.12	$\approx$ 3.7	$\approx$ 33	$\approx$ 50	$\approx$ 24	74.8 $\pm$ 0.6
FedInMM	$\approx$ 4.85	$\approx$ 1.43	$\approx$ 4.8	$\approx$ 41	$\approx$ 67	$\approx$ 29	79.7 $\pm$ 0.5
CreamFL	$\approx$ 4.75	$\approx$ 1.45	$\approx$ 4.8	$\approx$ 39	$\approx$ 64	$\approx$ 28	78.3 $\pm$ 0.6
PmcmFL	$\approx$ 4.80	$\approx$ 1.42	$\approx$ 4.7	$\approx$ 43	$\approx$ 69	$\approx$ 30	80.3 $\pm$ 0.4
FedMAC	$\approx$ 4.85	$\approx$ 1.44	$\approx$ 4.8	$\approx$ 42	$\approx$ 68	$\approx$ 30	79.9 $\pm$ 0.5
MFCPL	$\approx$ 0.27	$\approx$ 1.46	$\approx$ 4.6	$\approx$ 41	$\approx$ 66	$\approx$ 29	80.0 $\pm$ 0.5
MODALIS (our work)	$\approx$ 0.11	$\approx$ 1.35	$\approx$ 4.5	$\approx$ 44	$\approx$ 70	$\approx$ 31	83.4 $\pm$ 0.4

^a The results are reported as the average uplink payload per client message in each communication round.

between computational efficiency, communication cost, and model performance.

6.5. Ablation studies

To gain deeper insight into the behavior of MODALIS, this section presents a set of ablation and sensitivity studies focusing on: (i) the main architectural components and loss terms; (ii) the impact of the underlying communication topology; and (iii) the sensitivity of key hyperparameters governing the multi-objective loss, prototype updates, and embedding dimensionality, with the explicit goal of quantifying how robust MODALIS is to moderate changes in these settings and whether extensive per-scenario fine-tuning is required. Unless otherwise stated, all experiments are conducted on the AVMNIST dataset under moderate heterogeneity ($\beta = 0.5, p_m = 0.25$).

Ablation study of MODALIS components. The first study evaluates the contributions of the core architectural blocks and each term in the multi-objective loss of Eq. (4). Table 8 reports the resulting average Macro-F1 scores and relative computational overhead for variants in which a single component is removed or modified at a time.

Removing the Contextual Null Embedding module ($\mathcal{N}_M$) resulted in an absolute F1 score reduction of 47.1%, confirming that naive imputation of missing modalities severely harms performance even when all other components are kept unchanged. Eliminating the gating module ($\mathcal{G}$) or masked fusion F similarly results in large drops, indicating that both adaptive modality weighting and explicit masking of imputed or absent modalities are necessary to obtain robust multimodal representations. The adaptive EMA update for prototypes also plays an important role: replacing it with a fixed momentum degrades accuracy while slightly reducing per-round computation, showing that the small added complexity of the adaptive scheme yields a favorable accuracy-efficiency trade-off. Ablations on the loss function further reveal that

both $\mathcal{L}_{\text{FAL}}$ and $\mathcal{L}_{\text{DCL}}$ are critical under heterogeneous data, resulting in absolute drops of 28.1% and 39.8%, respectively. In addition, $\mathcal{L}_{\text{MCL}}$ and $\mathcal{L}_{\text{PR}}$ serve as lighter regularizers, providing consistent yet more modest improvements. Turning off the core prototype exchange mechanism and forcing clients to train in isolation with their prototypes resulted in a 35.1% drop, confirming that collaborative learning through prototype sharing is the cornerstone of success. These results strongly justify the comprehensive design of MODALIS, as the severe degradation in model performance vastly outweighs the minimal computational time savings gained from removing any single component. The study also validated the choice of a single local epoch ($T = 1$), as increasing T to five local epochs significantly harmed performance due to client drift, while drastically increasing computation time.

Topology sensitivity. To assess robustness with respect to the communication topology, MODALIS was evaluated under several decentralized graphs with the same number of clients. Table 9 compares the default static ER graph with edge probability $p_{ER} = 0.9$ to: (i) a fully connected graph; (ii) a dynamic ER graph with $p_{ER} = 0.9$ in which neighbors are resampled randomly at each round; (iii) a sparser static ER graph with $p_{ER} = 0.3$; (iv) a sparse dynamic ER graph with $p_{ER} = 0.3$; and (v) a ring topology with two neighbors per client.

Fully connected and dense ER graphs achieve the highest scores, as expected: the fully connected topology offers a small improvement to 90.5%, and dynamic ER with $p_{ER} = 0.9$ reaches 90.3% relative to the static ER baseline. Both static and dynamic sparse ER topologies incur only moderate degradation, with Macro-F1 dropping to 88.2% and 88.5%, despite their substantially lower average degree. The ring topology exhibits the largest drop (a 5.0-point decrease), reflecting the slower information mixing inherent to very sparse, low-diameter graphs. Importantly, the dynamic ER graphs perform comparably to their static counterparts at the same density, indicating that the prototype-based aggregation mechanism is robust to moderate topology dynamics and

Table 8

 Ablation study of MODALIS components using AVMNIST dataset ($\beta = 0.5, p_m = 0.25$).

Variant configuration	Macro-F1 (%)	Time per round (s)
MODALIS (full model)	90.1±0.2	81.0
<i>Architectural components</i>		
w/o Contextual Null Embed ($\mathcal{N}_M$)	43.0±1.5 ($\downarrow 47.1$)	78.3 ($\downarrow 3.3\%$)
w/o Gating Module ($\mathcal{G}$)	68.0±0.9 ($\downarrow 22.1$)	79.6 ($\downarrow 1.7\%$)
w/o Masked Fusion in $\mathcal{F}$	55.0±1.1 ($\downarrow 35.1$)	77.1 ($\downarrow 4.8\%$)
w/o Adaptive EMA (fixed $m_t = 0.9$)	79.0±0.7 ($\downarrow 11.1$)	72.9 ($\downarrow 10.0\%$)
<i>Loss components from Eq. (4)</i>		
w/o $\mathcal{L}_{\text{FAL}}$	62.0±1.0 ($\downarrow 28.1$)	78.3 ($\downarrow 3.3\%$)
w/o $\mathcal{L}_{\text{DCL}}$	50.3±1.2 ($\downarrow 39.8$)	74.3 ($\downarrow 8.3\%$)
w/o $\mathcal{L}_{\text{MCL}}$	87.4±0.4 ($\downarrow 2.7$)	79.6 ($\downarrow 1.7\%$)
w/o $\mathcal{L}_{\text{PR}}$	89.2±0.3 ($\downarrow 0.9$)	79.6 ($\downarrow 1.7\%$)
<i>Protocol variants</i>		
$T = 5$ local epochs per round	82.2±0.7 ($\downarrow 7.9$)	340.2 ($\uparrow 320\%$)
Local prototypes only (no exchange)	55.0±1.3 ($\downarrow 35.1$)	64.8 ($\downarrow 20.0\%$)

Table 9

 Impact of network topology on MODALIS using AVMNIST ($\beta = 0.5, p_m = 0.25$).

Topology configuration	Macro-F1 (%)	Time per round (s)
Static ER ($p_{\text{ER}} = 0.9$)	90.1±0.2	81.0
Fully connected	90.5±0.2 ($\uparrow 0.4$)	82.5
Dynamic ER ($p_{\text{ER}} = 0.9$)	90.3±0.3 ($\uparrow 0.2$)	81.4
Sparse static ER ($p_{\text{ER}} = 0.3$)	88.2±0.5 ($\downarrow 1.9$)	79.2
Sparse dynamic ER ($p_{\text{ER}} = 0.3$)	88.5±0.4 ($\downarrow 1.6$)	79.8
Ring (2 neighbors per client)	85.1±0.6 ($\downarrow 5.0$)	78.6

does not rely on a fixed neighbor set. These trends empirically support the theoretical dependence on the mixing properties of the communication matrix captured by ρ in Assumption 4. When scaling to large N , preserving a small-world or expander-like structure with bounded average degree keeps the spectral gap large and the effective diameter small, so prototype information continues to propagate efficiently even with hundreds or thousands of clients. In contrast, very sparse, chain-like topologies have diameters that grow linearly with N , and thus require substantially more communication rounds or additional mechanisms (e.g., hierarchical aggregation or occasional long-range links) to reach a comparable level of prototype consensus.

Hyperparameter sensitivity and representation capacity. A complementary sensitivity analysis investigates the hyperparameters that most directly control prototype learning and the multi-objective loss in Eq. (4). Table 10 reports the effect of varying: (i) the weights of the alignment and contrastive losses ($w_{\text{FAL}}, w_{\text{DCL}}$); (ii) the bounds of the adaptive EMA momentum ($m_{\min}, m_{\max}$); (iii) the $\mathcal{L}_{\text{DCL}}$ parameters (τ, μ, s); and (iv) the embedding dimensionality d of the shared representation and prototype space. In all cases, differences are reported relative to the common baseline configuration ($w_{\text{FAL}}=0.5, w_{\text{DCL}}=0.2, (m_{\min}, m_{\max})=(0.5, 0.99), (\tau, \mu, s)=(0.1, 0.3, 30), d=128$).

The results exhibit a broad and smooth plateau around the default configuration. For w_{FAL} , setting the weight to zero reduces Macro-F1 from 90.1% to 86.8%, and a small weight of 0.1 yields 87.5%. Increasing w_{FAL} to 1.0 slightly reduces performance to 89.8%, indicating that the method is not overly sensitive to moderate over-weighting as long as the task loss remains dominant. A similar pattern holds for w_{DCL} : removing the term altogether reduces Macro-F1 to 86.9%, while values in the range [0.05, 0.5] keep performance within 1.2 points of the baseline.

Table 10

 Sensitivity of MODALIS to key hyperparameters on AVMNIST ($\beta = 0.5, p_m = 0.25$).

Group	Setting	Macro-F1 (%)
Baseline	–	90.1±0.2
Loss weight w_{FAL}	$w_{\text{FAL}} = 0.0$	86.8±0.5 ($\downarrow 3.3$)
	$w_{\text{FAL}} = 0.1$	87.5±0.4 ($\downarrow 2.6$)
	$w_{\text{FAL}} = 0.5$	90.1±0.2
	$w_{\text{FAL}} = 1.0$	89.8±0.3 ($\downarrow 0.3$)
Loss weight w_{DCL}	$w_{\text{DCL}} = 0.0$	86.9±0.5 ($\downarrow 3.2$)
	$w_{\text{DCL}} = 0.05$	88.9±0.4 ($\downarrow 1.2$)
	$w_{\text{DCL}} = 0.2$	90.1±0.2
	$w_{\text{DCL}} = 0.5$	89.5±0.3 ($\downarrow 0.6$)
EMA bounds ($m_{\min}, m_{\max}$)	(0.1, 0.5)	88.1±0.5 ($\downarrow 2.0$)
	(0.3, 0.9)	89.4±0.3 ($\downarrow 0.7$)
	(0.5, 0.99)	90.1±0.2
	(0.7, 0.995)	89.2±0.4 ($\downarrow 0.9$)
DCL params (τ, μ, s)	(0.9, 0.999)	88.9±0.4 ($\downarrow 1.2$)
	(0.1, 0.3, 30)	90.1±0.2
	(0.07, 0.3, 30)	89.7±0.3 ($\downarrow 0.4$)
	(0.2, 0.3, 30)	89.1±0.3 ($\downarrow 1.0$)
Embedding dim. d	(0.1, 0.0, 30)	86.4±0.6 ($\downarrow 3.7$)
	(0.1, 0.5, 30)	89.5±0.4 ($\downarrow 0.6$)
	(0.1, 0.3, 50)	88.8±0.4 ($\downarrow 1.3$)
	$d = 64$	88.6±0.3 ($\downarrow 1.5$)
Embedding dim. d	$d = 96$	89.5±0.3 ($\downarrow 0.6$)
	$d = 128$	90.1±0.2
	$d = 256$	90.4±0.3 ($\uparrow 0.3$)

The EMA bounds ($m_{\min}, m_{\max}$) show a gradual effect: very conservative updates (0.1, 0.5) reduce Macro-F1 to 88.1%, while slightly tighter or looser ranges such as (0.3, 0.9) or (0.7, 0.995) remain within 0.7–0.9 points of the baseline. Extremely slow adaptation (0.9, 0.999) results in a drop to 88.9%. For the DCL parameters, removing the angular margin ($\mu = 0$) is particularly detrimental, decreasing Macro-F1 from 90.1% to 86.4%. Changing the temperature from 0.1 to 0.2, or slightly adjusting μ or s , results in much smaller drops (between 0.4 and 1.3 points), indicating

that the contrastive term is robust to moderate perturbations around the default.

Finally, the representation capacity experiments show that increasing the embedding dimension d from 64 to 128 improves Macro-F1 from 88.6% to 90.1%, whereas further increasing d to 256 yields only a marginal additional improvement to 90.4% at the cost of doubling the prototype dimensionality and hence the communication payload. This behavior is consistent with the communication-cost analysis in Lemma 1, which shows that beyond a certain point, additional embedding capacity yields diminishing returns relative to the linear growth in prototype size

7. Discussion

The empirical results presented in Section 6 strongly support the efficacy of MODALIS. This section contextualizes these findings by highlighting the framework's core contributions relative to the state of the art, analyzing its key trade-offs, and candidly addressing its limitations, thereby outlining clear directions for future research.

Novelty and core contributions: The primary contribution of MODALIS is the synthesis of a fully decentralized communication protocol with a sophisticated, multimodal-aware client architecture. Unlike prior art, which typically addresses decentralization and multimodal heterogeneity in isolation, MODALIS tackles both simultaneously. Its prototype-centric protocol directly circumvents the communication bottlenecks and knowledge dilution issues inherent to parameter-averaging DFL methods. Architecturally, the framework's novelty lies in its end-to-end approach to handling incomplete data. The Contextual Null Embedding module represents a significant design advance over static imputation, enabling intelligent, feature-conditioned generation of missing modality representations. This robust input processing is then complemented by an adaptive gating mechanism that dynamically weighs the importance of each modality, followed by a masked multi-way transformer for fusion, which explicitly prevents less reliable imputed data from corrupting the final representation. Furthermore, this entire process is guided by a multi-objective loss function that provides a principled mechanism for enforcing representation consistency, not only across the decentralized network ($\mathcal{L}_{\text{FAL}}$, $\mathcal{L}_{\text{DCL}}$) but also across modalities within a single client ($\mathcal{L}_{\text{MCL}}$). This synergy is what enables effective cross-modal knowledge transfer, a capability validated by the strong performance of unimodal clients (see Table 5).

Interpreting performance under heterogeneity: The resilience of MODALIS is most evident in high-stress scenarios ($\beta = 0.1$). As shown in Table 4, while the performance of all methods degrades with increasing heterogeneity, MODALIS consistently maintains the highest F1 scores. For example, on AVMNIST with $p_m = 0.5$, MODALIS achieves an F1 score of 83.4%, a significant +3.1% absolute gain over the next-best federated method, PmcMFL, and a 13.9% gain over the standard FedAvg. This superior performance can be directly attributed to the prototype-guided multi-objective loss function. By regularizing local training with aggregated prototypes from neighbors (via losses such as $\mathcal{L}_{\text{FAL}}$ and $\mathcal{L}_{\text{DCL}}$), MODALIS effectively counteracts client drift induced by local data biases, enabling the network to converge to a more robust and generalized solution. Furthermore, the outstanding performance of unimodal clients (Table 5) confirms the effectiveness of the cross-modal knowledge transfer mechanism. A client possessing only image data, for example, successfully leverages audio prototypes from its neighbors to regularize its training, a feat made possible only by the coherent cross-modal representations enforced by the framework's design.

Efficiency and decentralization trade-offs: The MODALIS prototype-exchange mechanism yields substantial communication savings, estimated at over $40\times$ compared to full model exchange methods (Table 7), which is a critical advantage for deployment in bandwidth-constrained environments. A noteworthy comparison is with FedProto, a centralized prototype-based method that is even more communication-efficient. The consistent underperformance of

FedProto relative to MODALIS highlights a key trade-off: MODALIS leverages decentralized, peer-to-peer communication to achieve a higher-quality consensus on prototypes, resulting in better accuracy with only a modest increase in communication overhead compared to server-centric prototype approaches.

Communication, computation, and privacy: From a system-design perspective, the prototype-centric protocol of MODALIS scales gracefully with the number of clients, since each device exchanges a fixed-size prototype library only with its neighbors, so the per-round communication cost scales with the node degree rather than the total network size. The convergence analysis in Section 5 makes this dependence explicit through the mixing parameter ρ . Moreover, the empirical study in Sections 6.4 and 6.5 shows that MODALIS maintains competitive wall-clock training time and moderate resource usage compared to recent multimodal FL baselines, despite its richer architecture and fusion modules. At the same time, Section 5.3 clarifies that the current design offers only intuitive privacy advantages over gradient sharing and does not provide formal guarantees, such as those provided by DP. In particular, clients with very few samples per class are more vulnerable to membership inference and related attacks, because their class prototypes are averaged over a small number of examples and may encode more sample-specific features.

Topology and convergence: Scaling to much larger networks (e.g., $N \geq 10^3$) primarily affects the number of rounds required for prototypes to mix globally, rather than the local computation or per-round communication budget of each client. For well-connected small-world or expander-like topologies, the graph diameter and mixing time grow only logarithmically with N , so the spectral gap and, consequently, the parameter ρ remain favorable. In this regime, prototype convergence degrades slowly as the network grows, at the cost of a modest increase in communication rounds. In contrast, very sparse or elongated graphs have large diameters and poorer mixing properties, which translate into larger consensus-error terms in Theorem 1 and empirically slower propagation of information across distant regions of the network, as confirmed by the topology-sensitivity study in Section 6.5 comparing dense ER graphs, sparser graphs, and a ring topology. For large deployments, degree-capped small-world overlays and hierarchical or clustered communication backbones can keep node degree and per-round communication bounded while preserving a small effective diameter and fast prototype convergence. For resource-constrained devices, lightweight variants of MODALIS with smaller backbones, lower-dimensional embeddings, or pruned auxiliary losses are a promising direction for low-power federated multimodal learning.

Limitations and Scope: Despite its strong performance, MODALIS has several limitations. First, the present study explores sparser and dynamic ER topologies. However, the experiments still assume a fixed set of participating clients and abstract away explicit client churn, physical node mobility, and realistic network latencies. This design isolates the impact of the proposed multimodal prototype-learning mechanisms from confounding networking effects. Moreover, while the neighbor-to-neighbor communication model scales favorably in terms of per-round cost, in extremely large or very sparsely connected networks, the number of rounds required to achieve a given level of global consensus may increase due to slower information mixing. In many real-world decentralized systems, such as mobile ad-hoc networks, clients may join or leave at any time, and links can become unreliable. Therefore, extending the framework to support dynamic topologies is an essential next step. Second, as noted in Section 5, prototype sharing offers stronger privacy than gradient sharing but falls short of formal guarantees, such as differential privacy. Adding rigorous privacy mechanisms will be necessary for applications that handle highly sensitive data. Third, the cross-dataset transfer analysis in Section 6.3.4 is intentionally limited to configuration-level transfer, where a single set of hyperparameters and architectural choices tuned on a source dataset is reused when training on other datasets. Fully zero-shot deployment of a model trained end-to-end on one dataset directly to another with a different label or

modality structure is not addressed in the current work, and more advanced forms of cross-domain adaptation, such as feature-level transfer or fine-tuning under distribution shift, remain an important direction for future research. Fourth, although the experimental protocol already includes challenging configurations with Dirichlet parameter $\beta = 0.1$ and missing-modality proportion $p_m = 0.5$, even more extreme regimes are expected to induce stronger degradation of representation quality and downstream performance. Addressing such pathological settings may require additional mechanisms, such as adaptive client selection, curriculum-style training schedules, or task-aware regularization, which lie beyond the scope of the present study and represent an important avenue for future work.

8. Conclusion

This paper presented MODALIS, a DFL framework engineered to optimize collaborative training on heterogeneous multimodal data, delivering significant performance gains while drastically reducing network consumption. By combining a communication-efficient prototype-exchange protocol with a sophisticated client-side architecture and a multi-objective loss function, MODALIS enables effective collaborative learning without a central server. The core innovations, including contextual null embeddings for missing modalities and prototype-guided alignment losses, were shown to be critical for achieving state-of-the-art performance. Empirical results across four datasets confirmed that MODALIS establishes a new benchmark for decentralized multimodal intelligence. It significantly outperforms existing methods by improving F1 scores by up to 4% under severe data heterogeneity, while simultaneously reducing communication costs by over 40x compared to standard parameter-averaging approaches.

Future research will proceed in several key directions. The primary focus will be on extending the framework to accommodate dynamic network topologies, enabling it to handle client churn and evolving connectivity. Another crucial avenue is to incorporate formal privacy guarantees and robustness to adversarial attacks, making the framework suitable for deployments with strict security requirements. Finally, investigating methods to enhance scalability for scenarios with extremely large numbers of classes or modalities and exploring the integration of fairness-aware mechanisms to ensure equitable performance will be important for broadening the practical impact of this work.

CRediT authorship contribution statement

Enrique Tomás Martínez Beltrán: Writing - original draft, Validation, Methodology, Investigation, Formal analysis, Data curation, Conceptualization; **G  r  me Bovet:** Resources, Project administration, Conceptualization; **Gregorio Mart  nez P  rez:** Supervision, Resources, Project administration, Funding acquisition; **Alberto Huertas Celdr  n:** Writing - review & editing, Validation, Supervision, Methodology, Conceptualization.

Data availability

Data will be made available on request.

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: Enrique Tomas Martinez Beltran reports financial support was provided by Fundaci  n S  neca, Regi  n de Murcia (Spain). Gregorio Martinez Perez reports financial support was provided by the Swiss Federal Office for Defense Procurement (armasuisse). Gregorio Martinez Perez reports financial support was provided by the European Defence Fund (EDF). If there are other authors, they declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work has been partially supported by (a) 21629/FPI/21, Fundaci  n S  neca, Regi  n de Murcia (Spain), and (b) the Swiss Federal Office for Defense Procurement (armasuisse) with the DECIMAL project. This work has also received funding from the European Union via the European Defence Fund project EU-GUARDIAN under grant number 101103044. Views and opinions expressed are, however, those of the author(s) only and do not necessarily reflect those of the European Union or the European Commission. Neither the European Union nor the granting authority can be held responsible for them.

References

- [1] B. McMahan, E. Moore, D. Ramage, S. Hampson, B.A.y. Arcas, Communication-efficient learning of deep networks from decentralized data, in: *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, 54 of *Proceedings of Machine Learning Research*, PMLR, 2017, pp. 1273–1282.
- [2] P. Kairouz, et al., Advances and open problems in federated learning, *Found. Trends  Mach. Learn.* 14 (1–2) (2021) 1–210. <https://doi.org/10.1561/22000000083>
- [3] E.T.M. Beltr  n, et al., Decentralized federated learning: fundamentals, state of the art, frameworks, trends, and challenges, *IEEE Commun. Surv. Tutor.* 25 (4) (2023) 2983–3013. <https://doi.org/10.1109/COMST.2023.3315746>
- [4] Y.-M. Lin, et al., Federated learning on multimodal data: a comprehensive survey, *Mach. Intell. Res.* 20 (4) (2023) 539–553. <https://doi.org/10.1007/s11633-022-1398-0>
- [5] H. Zhu, J. Xu, S. Liu, Y. Jin, Federated learning on non-IID data: a survey, *Neurocomputing* 465 (2021) 371–390. <https://doi.org/10.1016/j.neucom.2021.07.098>
- [6] L. Bai, H. Hu, Q. Ye, H. Li, L. Wang, J. Xu, Membership inference attacks and defenses in federated learning: a survey, *ACM Comput. Surv.* 57 (4) (2024). <https://doi.org/10.1145/3704633>
- [7] Y. Tan, G. Long, L.U. LIU, T. Zhou, Q. Lu, J. Jiang, C. Zhang, FedProto: federated prototype learning across heterogeneous clients, *Proc. AAAI Conf. Artif. Intell.* 36 (8) (2022) 8432–8440. <https://doi.org/10.1609/aaai.v36i8.20819>
- [8] X. Mu, Y. Shen, K. Cheng, X. Geng, J. Fu, T. Zhang, Z. Zhang, FedProc: prototypical contrastive federated learning on non-IID data, *Fut. Gen. Comput. Syst.* 143 (2023) 93–104. <https://doi.org/10.1016/j.future.2023.01.019>
- [9] T.-K. Tran, H.-P. Tran, T.-L. Le, T.-H. Tran, FedNTProto: a prototype-based approach for personalized federated learning, in: *2024 International Conference on Multimedia Analysis and Pattern Recognition (MAPR)*, 2024, pp. 1–6. <https://doi.org/10.1109/MAPR63514.2024.10660707>
- [10] L. Qi, H. Chen, H. Zou, S. Chen, X. Zhang, H. Chen, Decentralized federated learning with prototype exchange, *Mathematics* 13 (2) (2025). <https://doi.org/10.3390/math13020237>
- [11] B. Li, W. Gao, J. Xie, M. Gong, L. Wang, H. Li, Prototype-based decentralized federated learning for the heterogeneous time-varying IoT systems, *IEEE Internet Things J.* 11 (4) (2024) 6916–6927. <https://doi.org/10.1109/IJOT.2023.3313118>
- [12] H. Zhang, F. Xu, Z. Yu, S. Pang, C. Hu, J. Yu, DFPL: decentralized federated prototype learning across heterogeneous data distributions, *arXiv preprint arXiv:2505.04947* (2025).
- [13] J. Chen, A. Zhang, FedMBrige: bridgeable multimodal federated learning, in: *Proceedings of the 41st International Conference on Machine Learning*, 235 of *Proceedings of Machine Learning Research*, PMLR, 2024, pp. 7667–7686.
- [14] S. Wang, Z. Qu, Y. Liu, S. Kan, Y. Liang, J. Wang, FedMMR: multi-modal federated learning via missing modality reconstruction, in: *2024 IEEE International Conference on Multimedia and Expo (ICME)*, 2024, pp. 1–6. <https://doi.org/10.1109/ICME57554.2024.10688162>
- [15] R. Rahman, D.C. Nguyen, Multimodal federated learning with model personalization, in: *Optimization for ML Workshop, NeurIPS 2024*, 2024. Poster session, <https://neurips.cc/virtual/2024/101845>.
- [16] W. Huang, D. Wang, X. Ouyang, J. Wan, J. Liu, T. Li, Multimodal federated learning: concept, methods, applications and future directions, *Inf. Fusion* 112 (2024) 102576. <https://doi.org/10.1016/j.inffus.2024.102576>
- [17] G. Bao, Q. Zhang, D. Miao, Z. Gong, L. Hu, K. Liu, Y. Liu, C. Shi, Multimodal federated learning with missing modality via prototype mask and contrast, *arXiv preprint arXiv:2312.13508* (2023).
- [18] P. Poudel, P. Shrestha, S. Amgain, Y.R. Shrestha, P. Gyawali, B. Bhattarai, CAR-MFL: cross-modal augmentation by retrieval for multimodal federated learning with missing modalities, in: *Proceedings of Medical Image Computing and Computer Assisted Intervention – MICCAI 2024*, LNCS 15010, 2024, pp. 102–112.
- [19] T.H. Phung, B.P. Nguyen, T.H. Nguyen, Q. Viet Hung Nguyen, P. Le Nguyen, T.T. Huynh, A contrastive learning and graph-based approach for missing modalities in multimodal federated learning, in: *2024 International Joint Conference on Neural Networks (IJCNN)*, 2024, pp. 1–8. <https://doi.org/10.1109/IJCNN60899.2024.10650285>
- [20] Y. An, Y. Bai, Y. Liu, L. Guo, X. Chen, A multimodal federated learning framework for modality incomplete scenarios in healthcare, in: *Bioinformatics Research and Applications: 20th International Symposium, ISBRA 2024*, Kunming, China, July 19–21, 2024, *Proceedings, Part II*, 2024, p. 245–256. https://doi.org/10.1007/978-981-97-5131-0_21

- [21] S. Yu, J. Wang, W. Hussein, P.C.K. Hung, Robust multimodal federated learning for incomplete modalities, *Comput. Commun.* 214 (2024) 234–243. <https://doi.org/10.1016/j.comcom.2023.12.003>
- [22] Q. Yu, Y. Liu, Y. Wang, K. Xu, J. Liu, Multimodal federated learning via contrastive representation ensemble, *arXiv preprint arXiv:2302.08888* (2023).
- [23] K. Yin, Z. Ding, X. Ji, Z. Wang, Self-attention fusion and adaptive continual updating for multimodal federated learning with heterogeneous data, *Neural Netw.* 187 (2025) 107345. <https://doi.org/10.1016/j.neunet.2025.107345>
- [24] H. Q. Le, C. M. Thwal, Y. Qiao, Y. L. Tun, M. N. H. Nguyen, E.-N. Huh, C.S. Hong, Cross-modal prototype based multimodal federated learning under severely missing modality, *Inf. Fusion* 122 (2025) 103219. <https://doi.org/10.1016/j.inffus.2025.103219>
- [25] S.P. Karimireddy, S. Kale, M. Mohri, S. Reddi, S. Stich, A.T. Suresh, SCAFFOLD: stochastic controlled averaging for federated learning, in: *Proceedings of the 37th International Conference on Machine Learning*, 119 of *Proceedings of Machine Learning Research*, PMLR, 2020, pp. 5132–5143.
- [26] Y.-H.H. Tsai, S. Bai, P.P. Liang, J.Z. Kolter, L.-P. Morency, R. Salakhutdinov, Multimodal transformer for unaligned multimodal language sequences, in: *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 2019, pp. 6558–6569. <https://doi.org/10.18653/v1/P19-1656>
- [27] T.-Y. Lin, P. Goyal, R. Girshick, K. He, P. Dollár, Focal loss for dense object detection, *IEEE Trans. Pattern Anal. Mach. Intell.* 42 (2) (2020) 318–327. <https://doi.org/10.1109/TPAMI.2018.2858826>
- [28] J. Deng, J. Guo, N. Xue, S. Zafeiriou, Arcface: additive angular margin loss for deep face recognition, in: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 4690–4699.
- [29] X. Lian, C. Zhang, H. Zhang, C.-J. Hsieh, W. Zhang, J. Liu, Can decentralized algorithms outperform centralized algorithms? a case study for decentralized parallel stochastic gradient descent, in: I. Guyon, U.V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, R. Garnett (Eds.), *Advances in Neural Information Processing Systems*, 30, 2017, pp. 5336–5346.
- [30] A. Koloskova, S. Stich, M. Jaggi, Decentralized stochastic optimization and gossip algorithms with compressed communication, in: K. Chaudhuri, R. Salakhutdinov (Eds.), *Proceedings of the 36th International Conference on Machine Learning*, 97 of *Proceedings of Machine Learning Research*, PMLR, 2019, pp. 3478–3487.
- [31] L. Zhu, Z. Liu, S. Han, Deep leakage from gradients, in: *Proceedings of the 33rd International Conference on Neural Information Processing Systems (NeurIPS 2019)*, 32, 2019, pp. 14774–14784.
- [32] C. Dwork, A. Roth, The algorithmic foundations of differential privacy, *Found. Trends Theor. Comput. Sci.* 9 (3-4) (2014) 211–407. <https://doi.org/10.1561/04000000042>
- [33] S. Becker, J. Vielhaben, M. Ackermann, K.-R. Müller, S. Lapuschkin, W. Samek, AudioMNIST: exploring explainable artificial intelligence for audio analysis on a simple benchmark, *J. Franklin Inst.* 361 (1) (2024) 418–428. <https://doi.org/10.1016/j.jfranklin.2023.11.038>
- [34] H. Cao, D.G. Cooper, M.K. Keutmann, R.C. Gur, A. Nenkova, R. Verma, CREMA-D: crowd-sourced emotional multimodal actors dataset, *IEEE Trans. Affect. Comput.* 5 (4) (2014) 377–390. <https://doi.org/10.1109/TAFFC.2014.2336244>
- [35] D. Anguita, A. Ghio, L. Oneto, X. Parra, J.L. Reyes-Ortiz, et al., A public domain dataset for human activity recognition using smartphones, in: *Proceedings of the 21th International European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning*, 3, 2013, pp. 3–4.
- [36] F. Alam, F. Ofli, M. Imran, CrisisMMD: multimodal twitter datasets from natural disasters, *Proc. 12th Int. AAAI Conf. Web Soc. Media (ICWSM)* 12 (1) (2018). <https://doi.org/10.1609/icwsm.v12i1.14983>
- [37] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 770–778. <https://doi.org/10.1109/CVPR.2016.90>
- [38] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, Bert: pre-training of deep bidirectional transformers for language understanding, in: *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, 2019, pp. 4171–4186.
- [39] T.-M.H. Hsu, H. Qi, M. Brown, Measuring the effects of non-identical data distribution for federated visual classification, *arXiv preprint arXiv:1909.06335* (2019).
- [40] E.T.M. Beltrán, G. Bovet, G.M. Pérez, A.H. Celdrán, NEBULA - Decentralized Federated learning for heterogeneous networks, in: *Proceedings of the ACM SIGCOMM 2025 Posters and Demos, ACM SIGCOMM Posters and Demos '25 2025*, pp. 149–151. <https://doi.org/10.1145/3744969.3748413>

