Mälardalen University
Department of Computer science and Engineering
Västerås, Sweden

Master of Science in Computer Science and Engineering , 30 credits

AGGREGATION OF EXPLANATIONS ACROSS 5G NETWORK DOMAINS USING AGGREGATION METRICS

Alexis Gbeckor-Kove
age25001@student.mdh.se

Examiner: Daneshtalab Masoud
Mälardalen University, Västerås, Sweden

Supervisors: Shaibal Barua
Mälardalen University, Västerås, Sweden

Company Supervisors: Vandita Singh, Ahmad Terra
Ericsson AB, Stockholm, Sweden

August 24, 2026

This thesis was conducted within the Erasmus Mundus Joint Master Degree Programme on the Engineering of Data-intensive Intelligent Software Systems (EDISS).

Abstract

As fifth-generation (5G) networks grow in complexity and increasingly incorporate AI-driven autonomous management, the ability to explain automated decisions has become critical for operator trust, regulatory compliance, and network governance. Explainable Artificial Intelligence (XAI) methods, and in particular SHAP (SHapley Additive exPlanations), provide principled feature attribution scores for individual predictive models. However, in multi-domain 5G architectures where distinct machine-learning models predict Quality of Service (QoS) KPIs across the Radio Access Network (RAN), Core, and Transport domains, explanations are inherently model-specific, operate over partially heterogeneous feature subsets, and are of unequal reliability. While prior work has proposed aggregation methods for feature-based explanations, no existing framework simultaneously addresses quality-informed weighting, formal aggregation across heterogeneous feature spaces, and alignment with operator-facing intents in a multi-domain 5G setting.

This thesis proposes a reliability-aware framework for aggregating SHAP feature attributions across heterogeneous 5G domain models. Central to the framework is the *Explanation Aggregation Metric* (EAM), which scores each model's explanation quality along three dimensions: *Fidelity*, *Stability*, and *Consistency*. These scores drive an EAM-weighted convex aggregation along the *cross-KPI* axis, which combines the different-KPI models of a single domain into a domain-level attribution profile. Along the complementary *cross-domain* axis, the per-domain models of a single KPI are instead composed through that KPI's physical QoS operator (additive for latency, minimum for throughput, and product for availability) rather than by reliability weighting, since the domains are related by a known composition law; this composition layer produces intent-aligned end-to-end explanations.

Evaluated on gradient-boosted tree models trained across three network domains and three KPIs, EAM weighting attenuates contributions from less reliable models in proportion to their measured explanation quality, without discarding them entirely, and top-ranked features remain consistent across both weighting schemes, confirming that dominant signals are robust to the choice of aggregation. Cross-model-family agreement on feature rankings within each domain, together with cross-domain agreement on the physical relevance of load-related features, provides empirical evidence that the framework recovers genuine network structure. The thesis identifies limitations of the current EAM formulation, most notably the absence of range-normalisation in the Stability score and the sampling-based nature of KernelSHAP attributions for composite operators, and outlines probable future works.

Table of Contents

List of Abbreviations **v**

List of Figures **vii**

1 Introduction **1**
1.1 Context and Motivation . . . 1
1.2 Intent-Based Networking and the Rise of Autonomy . . . 1
1.3 Explainable AI and its Current Limits . . . 2
1.4 Thesis Objective and Contributions . . . 2
1.5 The Multi-Domain Explanation Problem . . . 4
1.6 Proposed Framework . . . 4
1.7 Research Questions . . . 5

2 Background **6**
2.1 Intent-Based Networking . . . 6
2.2 Explainable AI and Shapley-Value Attribution . . . 8
2.3 5G Network Architecture and Functional Domains . . . 10
2.4 QoS Key Performance Indicators . . . 12

3 Related Work **14**
3.1 Explanation Quality Evaluation . . . 14
3.2 Explanation Aggregation Across Models . . . 15
3.3 XAI for 5G and O-RAN . . . 15
3.4 Intent-Based Explainability Orchestration . . . 16
3.5 Positioning of This Work . . . 16

4 Problem Formulation **17**
4.1 System Overview . . . 17
4.2 Problem Statement . . . 17
4.3 Formal Problem Definition . . . 18
4.4 Scope and Assumptions . . . 18

5 Methodology **20**
5.1 Data Generation via ns-3 Simulation . . . 20
5.1.1 Simulation Setup . . . 20
5.1.2 Parameter Sweep . . . 20
5.1.3 Feature Engineering . . . 20
5.1.4 Train/Validation/Test Split . . . 21
5.2 Model Training and Selection . . . 21
5.3 Feature Attribution Computation . . . 22
5.4 Aggregation Metric . . . 22
5.4.1 Overview . . . 22
5.4.2 Fidelity . . . 22
5.4.3 Stability . . . 23
5.4.4 Consistency . . . 23
5.5 Aggregation Algorithm . . . 24
5.6 QoS Algebra and Intent-Based Extension . . . 27

5.6.1 Throughput . . . 27
5.6.2 Availability . . . 28
5.6.3 Latency . . . 28
5.6.4 KernelSHAP for Composites . . . 29

6 System Design and Implementation 30
6.1 Data Collection Pipeline . . . 31
6.2 Multi-Domain Model Training . . . 31
6.3 Domain Model Interface . . . 31
6.4 Cross-Domain Orchestration . . . 32
6.5 KPI Composition and Explanation Strategies . . . 32
6.6 EAM-Weighted Aggregation . . . 34

7 Results 35
7.1 Model Performance . . . 35
7.2 Explanation Aggregation Metric Scores . . . 35
7.3 Per-(Domain, KPI) SHAP Attributions . . . 36
7.4 Cross-KPI Aggregation (per Domain) . . . 37
7.5 Cross-Domain Aggregation (per KPI) . . . 38
7.6 KPI-Composition Explanations . . . 39
7.7 Local Accuracy Check . . . 40
7.8 Ontology-Based Representation of the Aggregated Explanations . . . 40

8 Discussion 42
8.1 Interpretation of EAM Scores . . . 42
8.2 Weighted vs. Uniform Aggregation . . . 43
8.3 Triangulation: Cross-Model and Cross-Domain Agreement . . . 44
8.4 QoS-Composition Explanations . . . 44
8.5 Relation to Prior Work . . . 45
8.6 Answering the Research Questions . . . 45
8.7 Validation Design and Evidentiary Basis . . . 47
8.8 Strengths and Trade-offs of the Approach . . . 47
8.9 Transferability to Other Application Domains . . . 48
8.10 Limitations and Threats to Validity . . . 49

9 Conclusions and Future Work 51
9.1 Key Contributions . . . 51
9.2 Future Work . . . 52
9.3 Closing Remarks . . . 54

10 Research Ethics and Data Confidentiality 55
10.1 Ethical Implications of the Research Outcomes . . . 55

References 60

Appendix A Complete Per-Domain Feature Sets 61

Appendix B Per-(Domain, KPI) SHAP Attribution Plots 65

List of Abbreviations

Abbreviation	Definition
3GPP	3rd Generation Partnership Project
5G	Fifth Generation (mobile networks)
5GC	5G Core Network
6G	Sixth Generation (mobile networks)
AI	Artificial Intelligence
AMF	Access and Mobility Management Function
BH	Backhaul
CatBoost	Categorical Boosting
CQI	Channel Quality Indicator
DeepSHAP	Deep SHapley Additive exPlanations
DSR	Design Science Research
E2E	End-to-End
EAM	Explanation Aggregation Metric
eMBB	enhanced Mobile Broadband
ETSI	European Telecommunications Standards Institute
EU AI Act	European Union Artificial Intelligence Act
gNB	next-generation NodeB (5G base station)
HARQ	Hybrid Automatic Repeat reQuest
IBN	Intent-Based Networking
IETF	Internet Engineering Task Force
IMF	Intent Management Function
KernelSHAP	Kernel SHapley Additive exPlanations
KPI	Key Performance Indicator
LIME	Local Interpretable Model-agnostic Explanations
LightGBM	Light Gradient Boosting Machine
mMTC	massive Machine-Type Communications
NR	New Radio
ns-3	Network Simulator 3
O-RAN	Open Radio Access Network
PII	Personally Identifiable Information
QoS	Quality of Service
RAN	Radio Access Network
rApp	non-real-time RAN Intelligent Controller application (O-RAN)
RFC	Request for Comments
RMSE	Root Mean Squared Error

continued on next page

Abbreviation	**Definition**
R^2	Coefficient of Determination
SHAP	SHapley Additive exPlanations
SINR	Signal-to-Interference-plus-Noise Ratio
SMF	Session Management Function
TreeSHAP	Tree SHapley Additive exPlanations
UE	User Equipment
UPF	User Plane Function
URLLC	Ultra-Reliable Low-Latency Communications
XAI	Explainable Artificial Intelligence
XGBoost	eXtreme Gradient Boosting
xApp	near-real-time RAN Intelligent Controller application (O-RAN)
ZSM	Zero-touch network and Service Management

List of Figures

1 5G network architecture and its three functional domains 11
2 System pipeline. Simulation data are collected via ns-3, gradient-boosted tree models are trained for each (domain, KPI) pair, and attributions are computed via TreeSHAP (per-domain models) or KernelSHAP (composite predictors). The EAM-weighted aggregation stage produces one global feature importance profile Φ per domain (RAN, Transport, Core, and E2E). 30
3 Cross-KPI aggregated feature importance (RAN and Core) 37
4 Cross-domain additive composition of latency 38
5 QoS-algebra composites explained with KernelSHAP 39
6 Throughput minimum routing at a single instance 40
7 SHAP beeswarm plots for the best RAN model per KPI. 65
8 SHAP beeswarm plots for the best Transport model per KPI. 65
9 SHAP beeswarm plots for the best Core model per KPI. 66
10 SHAP beeswarm plots for the best E2E model per KPI 66
11 SHAP feature-importance comparison across model families 67

1 Introduction

Telecommunication networks are undergoing a major transformation driven by large-scale virtualisation, cloud-native architectures, and data-driven automation. Modern network infrastructures increasingly rely on software-defined networking, network function virtualisation, and artificial intelligence to manage growing traffic volumes, heterogeneous services, and stringent quality-of-service requirements. Machine learning techniques are now widely applied to tasks such as traffic forecasting, anomaly detection, fault prediction, resource optimisation, and closed-loop control.

As network management becomes more complex and dynamic, traditional rule-based and manually operated approaches are no longer sufficient. To address this challenge, Intent-Based Networking (IBN) has emerged as a promising paradigm for automated network management. In IBN, high-level business or operational objectives, referred to as *intents*, are specified by operators or applications and interpreted by an Intent Management Function (IMF), which translates them into low-level policies, configurations, and control actions across the network infrastructure.

1.1 Context and Motivation

The fifth generation of mobile telecommunications (5G) represents a fundamental shift in how networks are architected, operated, and managed. Unlike its predecessors, 5G is designed not only to deliver higher peak data rates, but to simultaneously serve heterogeneous use cases with strict Quality of Service (QoS) requirements. Meeting these requirements at scale, and in a cost-effective manner, requires a departure from manual, rule-based network management towards *autonomous, AI-driven operation* [1]. Modern network infrastructures increasingly rely on network function virtualisation and machine learning to manage growing traffic volumes, heterogeneous services, and stringent QoS guarantees.

1.2 Intent-Based Networking and the Rise of Autonomy

This vision of autonomous operation is realised through multi-layer, *intent-based management* architectures. Intent-Based Networking (IBN) allows operators to express high-level business or operational objectives as *intents*,for example, “ensure latency below 10-ms for all UEs in sector A” without specifying how those objectives should be achieved. These intents are submitted to an Intent Management Framework (IMF), which interprets them and orchestrates their realisation across the Radio Access Network (RAN), Transport, and Core domains [2]. As networks progress toward higher levels of autonomy, particularly autonomy levels L4 and above [3], the translation of intents into concrete configurations is increasingly delegated to interconnected software components and learning-based models. While this delegation improves operational efficiency and scalability, it also reduces direct human oversight over decisions that affect service continuity, safety, and compliance. The deployment of AI in critical network infrastructure therefore raises an urgent accountability challenge. When an AI component misclassifies the network state, misconfigures a resource, or issues an incorrect intent recommendation, human operators must be able to understand why that specific decision was reached, which input features drove it, which domain contributed most, and whether the underlying model behaviour can be trusted in the current operating conditions. This

requirement for transparency is not merely operational: it is increasingly codified in regulatory frameworks such as the EU AI Act [4], which mandates explainability for high-risk automated systems. Operators, service providers, and regulators consequently demand visibility into how automated decisions are produced, which constraints and signals influence them, and what residual uncertainties or risks are involved.

1.3 Explainable AI and its Current Limits

Explainable Artificial Intelligence (XAI) addresses this need by providing post-hoc explanations for model predictions [5]. Among the most widely adopted XAI methods is SHapley Additive exPlanations (SHAP) [6], which assigns each input feature a score quantifying its contribution to a model's output. Since this thesis uses gradient-boosted tree models throughout, TreeSHAP [7] is the primary explanation method employed: it computes exact Shapley values for tree-based models in polynomial time, making it practical for the multi-domain, multi-model setting developed in this work. Complementary techniques such as Local Interpretable Model-Agnostic Explanations (LIME) [8], Integrated Gradients [9], and counterfactual explanations[10] offer further perspectives on individual model behaviour. However, in an IBN environment, a single network decision rarely originates from a single model. Instead, it emerges from the coordinated interaction of multiple heterogeneous AI components distributed across several domains such as RAN, Transport, and Core, each with its own input space, explanation method, and confidence profile. As a result, **model-level explanations alone are insufficient to account for system-level behaviour**. Recent research has begun to explore the aggregation of multiple explanation methods to improve robustness and reliability at the model level, but the integration and reconciliation of heterogeneous explanations *across distributed network management components* remains under explored. The absence of unified aggregation and traceability mechanisms limits operators' ability to validate automated actions, assess regulatory compliance, and diagnose unexpected behaviour.

1.4 Thesis Objective and Contributions

The gap this thesis addresses is therefore the following: while individual explanation methods, and even the aggregation of explanations *within* a single model, are relatively mature, there is no principled mechanism for *aggregating and reconciling heterogeneous, per-domain explanations into a single, reliability-aware account of a system-level, intent-driven decision.* This thesis addresses that gap by investigating the design of a *system-level explainability framework* for intent-based network management in 5G and beyond. The proposed approach is a *reliability-aware feature-attribution aggregation framework*: heterogeneous per-domain SHAP explanations are not equally trustworthy, so instead of being naively averaged they are normalised, scored for explanation quality, and combined in proportion to that quality, yielding an aggregated explanation that crosses component boundaries and can be tied back to operator-facing intents. Methodologically, the work follows the Design Science Research (DSR) paradigm [11], whose central concern is the construction and evaluation of an artefact that solves a relevant real-world problem, organised around the three DSR cycles. The *relevance* cycle grounds the artefact in

the operational needs of intent-based 5G management: the problem is drawn from the multi-domain, multi-model structure of real network operations (Section 2) and formalised as a set of requirements on the aggregated explanation (Section 4). The *rigour* cycle draws on the existing knowledge base, Shapley-value attribution theory together with prior explanation-quality and explanation-aggregation research (Section 3), and contributes the EAM metric back to it. The *design* cycle then iterates between building the artefact, the aggregation framework and its prototype orchestration engine (Sections 5 and 6), and evaluating it experimentally on realistic multi-domain telecom KPI data (Sections 7 and 8). The intended audience of this work includes network operators responsible for managing and maintaining network performance; system engineers designing and deploying AI-driven automation components; researchers working in network automation and explainable AI; and decision-makers responsible for governance and regulatory compliance.

Beyond the aggregation framework evaluated here, an ontology-based representation of intent- and model-level explanations was also designed and implemented within the wider industrial project in which this thesis was conducted. At a conceptual level, the ontology component outputs a machine-readable representation that expresses intents, the models and features that inform them, and the resulting explanations in a common vocabulary, so that explanations originating from heterogeneous domain models can be described, linked, and queried uniformly instead of existing as disconnected per-model artefacts. The concrete schema and vocabulary behind these outputs are proprietary and cannot be disclosed; the component is therefore summarised here for completeness but falls outside the scope of the contributions presented and evaluated in this thesis, which centre on the reliability-aware aggregation framework and its Explanation Aggregation Metric. The main contributions of this thesis are:

- a **reliability-aware SHAP aggregation framework** for multi-domain 5G networks that combines L1-normalised per-model attributions along two complementary axes: an EAM-weighted convex combination across KPIs within a domain, and a QoS-algebra composition (additive, minimum, product) across domains for a fixed KPI;

- a formal **Explanation Aggregation Metric (EAM)** that scores explanation quality along three operationalised dimensions, Fidelity, Stability, and Consistency, each bounded in $[0, 1]$;

- a **multi-domain 5G simulation dataset** spanning the RAN, Transport, Core, and E2E views, with an engineered load-ratio and interaction feature set;

- a **prototype orchestration engine** that composes per-domain predictions and explanations via the QoS algebra, matching the attribution strategy to each operator (exact TreeSHAP where the composition is additive or routed, KernelSHAP where the composite is treated as a black box).

- an **empirical validation** showing that the framework recovers physically interpretable cross-domain drivers and that dominant feature signals are robust to the choice of weighting scheme.

1.5 The Multi-Domain Explanation Problem

A significant yet underexplored challenge in 5G network XAI is the *multi-domain, multi-model* nature of the system. In this thesis, separate predictive models are trained for each network domain (RAN, Transport, Core, and E2E) and for each KPI of interest. This per-domain, per-KPI structure reflects the heterogeneous data, distinct feature spaces, and differing operational dynamics that characterise each network segment. Explanations from these models are therefore inherently fragmented: each model's attribution vector describes feature importance from its own narrow perspective, using its own scale and feature subset.

Naively aggregating these attribution vectors, for instance, by averaging them suffers from three fundamental problems:

1. **Scale incomparability**: attribution magnitudes differ across models because KPIs have different ranges and units (Mbps vs. milliseconds vs. percentage).
2. **Unequal reliability**: some models produce explanations that are more stable and faithful than others; averaging without accounting for quality can contaminate the aggregate with noise.
3. **Feature subset heterogeneity**: different domains use different feature sets, requiring careful zero-padding and normalisation before any cross-domain aggregation.

A principled aggregation method that accounts for these issues is essential for building trust in explanation-driven network management.

1.6 Proposed Framework

This thesis proposes a **reliability-aware feature attribution aggregation framework** tailored to multi-domain 5G networks. The core insight is that not all model explanations are equally trustworthy: an explanation produced by a stable, well-calibrated predictive model should carry more weight in the aggregate than one from a model whose attributions fluctuate under small input changes or fail to faithfully reflect the model's output.

To operationalise this insight, the framework introduces a *Explanation Aggregation Metric* (EAM) score, inspired by [12] for each model, which comprises multiple quality metrics such as:

- **Fidelity**: the degree to which masking the top-attributed features degrades model predictions, measuring how faithfully the explanation captures the model's dependence on those features [13].
- **Stability**: the robustness of attributions under small Gaussian input perturbations, following the framework of Alvarez-Melis and Jaakkola [14].
- **Consistency**: the mean pairwise cosine similarity of explanation vectors across test instances, capturing whether the model applies coherent feature weightings across diverse inputs. [15]

The framework is further extended with a *QoS algebra* layer that enables intent-based network management: composite predictors combine domain-model predictions using minimum, soft-minimum, and product operators to estimate E2E KPI

satisfaction, and KernelSHAP explanations are computed for these black-box composites.

1.7 Research Questions

This work is guided by the following research questions:

RQ1 How can feature attributions from multiple heterogeneous 5G domain models be aggregated into a coherent, globally interpretable feature importance representation?

RQ2 How can the reliability of individual model explanations be quantified, and how should this reliability be used to weight the aggregation?

RQ3 To what extent does reliability-aware aggregation produce higher-quality and more stable feature importance estimates compared to naive uniform averaging?

2 Background

This section establishes the technical foundations on which the remainder of the thesis rests. Section 2.1 introduces Intent-Based Networking (IBN), the management paradigm that provides the operational context for the explainability framework developed in Section 5. Section 2.2 then surveys the relevant Explainable AI (XAI) literature, focusing on Shapley-value attribution methods that underpin the technical approach. Section 2.3 introduces the 5G network architecture, decomposing it into the three functional domains, Radio Access Network (RAN), Transport, and Core, and defines the End-to-End (E2E) view that integrates them. Finally, Section 2.4 defines the three Quality-of-Service (QoS) Key Performance Indicators (KPIs) used throughout the thesis and motivates why each requires domain-specific treatment.

2.1 Intent-Based Networking

Intent-Based Networking (IBN) abstracts network management from low-level device configuration into high-level *intents*: declarative, goal-oriented statements of desired network behaviour [16, 17, 1]. The IETF has formalised the concept in RFC 9315, which defines an intent as "a set of operational goals (that a network should meet) and outcomes (that a network is supposed to deliver) defined in a declarative manner without specifying how to achieve or implement them" [16]. Rather than instructing the network *how* to achieve a configuration, an operator specifies *what* outcomes the network should deliver, and the management system is responsible for translating those outcomes into concrete actions and for maintaining them under changing conditions [17, 16]. A representative intent might read: "maintain throughput above 10 Mbps with 99% availability for all UEs in cell 3 during business hours."

IBN is closely tied to the broader move toward autonomous and zero-touch network operation. The ETSI Zero-Touch Network and Service Management (ZSM) reference architecture [18] formalises this direction by specifying closed-loop management domains in which goals, telemetry, analytics, and actuation are coordinated with minimal human intervention. Ericsson's autonomous-networks roadmap positions intent-driven operation as a principal abstraction through which higher levels of network autonomy are reached [1], a direction echoed by industry maturity models charting the autonomous-network journey [3]. In the 5G context specifically, the radio-access architecture and its interfaces to the core defined in 3GPP NR specifications [19] and the requirements documented for next-generation access technologies [20] establish the heterogeneous, multi-domain substrate over which such intents must be realised. Intent-driven operation has itself been standardised within 3GPP, which specifies intent-driven management services for mobile networks and defines the information model, lifecycle, and reporting mechanisms used by intent-aware management functions [21].

An Intent Management Function (IMF) is responsible for (i) parsing and decomposing the intent into domain-specific sub-goals, (ii) delegating those sub-goals to per-domain controllers and optimisation engines, and (iii) continuously verifying that the resulting network state satisfies the specified constraints, triggering remediation if not [16, 17, 21]. This closed-loop assurance step is what distinguishes IBN

from earlier policy-based management paradigms: the network is expected not only to act on the intent but also to autonomously detect and recover from drift away from it [18, 17].

Increasingly, the intelligence that drives these steps is supplied by machine learning. The scale, dynamism, and heterogeneity of 5G make hand-crafted rules brittle, so the IMF and its per-domain controllers increasingly rely on learned models, rather than fixed policies alone, to carry out the core functions of the intent loop [1, 3]. Predictive models estimate whether a candidate configuration will satisfy an intent's KPI constraints *before* deployment; regression and classification models map configuration and telemetry to expected performance; optimisation and reinforcement-learning agents search for configurations that realise an intent; and anomaly-detection models flag drift away from it during assurance. It is this reliance on data-driven components that leads such systems to be described as *AI-native* [2], and it is also what raises the accountability requirement developed below: because the decomposition, actuation, and assurance stages are carried out by models, not by transparent rules, operators must be able to understand and trust why a given model-driven decision was reached.

IBN therefore fundamentally relies on the ability to predict whether a given network configuration will satisfy intent constraints *before* the configuration is deployed, and to explain *why* a constraint is predicted to be violated so that operators and automated controllers can take targeted corrective action. This dual requirement, prediction and attribution, is precisely the problem addressed by the framework developed in this thesis. Recent work has begun to make explainability itself a first-class concern of the intent loop: Bilgin et al. [2] propose an end-to-end explainability orchestration framework for AI-native networks in which intent-driven control is paired with explanation services across the resource, service, and business levels, and Brik et al. [22] survey explainable-AI techniques specifically for 6G O-RAN deployments, identifying the interoperability and privacy of distributed XAI across a multi-vendor O-RAN among the open challenges.

In operational IBN environments, decisions emerge from the coordinated interaction of heterogeneous modules: intent parsers, policy reasoners, optimisation engines, learning-based predictors, and network actuators, each contributing partial reasoning and contextual information at different levels of the network hierarchy [17, 18]. This distributed decision-making makes system-level explainability both necessary and difficult: a single-domain explanation is insufficient because the root cause of an intent violation may lie in a different domain from the one in which the violation is observed.

In this thesis, intent satisfaction is modelled through *QoS algebra*: composite functions that combine per-domain KPI predictions using minimum, soft-minimum (experimental), and product operators to estimate the probability that end-to-end KPI thresholds are met. These composites serve as building blocks for intent verification and are described formally in Section 5.6. The explainability framework then attributes the composite prediction back to individual input parameters, providing the cross-domain root-cause signal required for IBN remediation.

2.2 Explainable AI and Shapley-Value Attribution

The widespread adoption of complex machine-learning models, particularly ensemble methods and deep neural networks, has created a tension between predictive performance and human interpretability. Explainable AI (XAI) has emerged as the research area concerned with resolving this tension by providing techniques that make model behaviour intelligible to human stakeholders, supporting debugging, validation, regulatory compliance, and end-user trust [5, 23]. Beyond technical motivation, regulatory frameworks such as the EU Artificial Intelligence Act [4] now place explicit transparency and explanation obligations on high-risk AI systems, giving XAI a normative as well as a scientific role.

Taxonomy of XAI techniques. Building on the taxonomy of Barredo Arrieta et al. [23] and the broader XAI literature, methods are commonly classified along several axes: *intrinsic* versus *post-hoc* (whether interpretability is a property of the model itself or produced by an external analysis after training); *model-specific* versus *model-agnostic* (whether the method exploits internal model structure or treats the model as a black box); and *local* versus *global* (whether the explanation describes a single prediction or the model as a whole). Within this taxonomy, several broad families of techniques are relevant to the present work:

- **Feature attribution methods**, which assign each input feature a numerical contribution to a particular prediction. Prominent examples are LIME [24], which fits a local linear surrogate around the instance to be explained; SHAP [6], which grounds attributions in cooperative game theory; and gradient-based methods for deep networks such as Integrated Gradients [9].
- **Counterfactual explanations**, which describe the minimal change to the input that would alter the model's decision [10]. These are particularly suited to actionable settings such as network remediation, where the operator wishes to know what configuration change would restore intent satisfaction. Mothilal et al. [25] further show that feature attribution and counterfactual explanations can be viewed as different means to the same underlying explanatory end.
- **Surrogate models**, which approximate a complex model with an interpretable one (e.g. a sparse linear model or shallow tree) either globally or in a local region around an instance, of which LIME is the canonical example [24].
- **Ontological and structured representations of explanations**, such as the Explanation Ontology of Chari et al. [26], which aim to provide a semantic layer that supports user-centred reasoning across these families and enables explanations to be represented and queried instead of being treated as opaque numerical outputs.

Shapley values. Among feature-attribution methods, the Shapley value [27] from cooperative game theory has emerged as a particularly principled choice because it is the *unique* attribution that satisfies four desirable axioms: *efficiency* (attributions sum to the difference between the prediction and a baseline), *symmetry* (features that contribute equally receive equal attribution), *dummy* (a feature that never changes the prediction receives zero attribution), and *additivity* (attributions are

linear over sums of games) [27]. For a model f and an instance x with feature set $N = \{1, \ldots, n\}$, the Shapley value of feature i is

$$\phi_i(f, x) = \sum_{S \subseteq N \setminus \{i\}} \frac{|S|!\,(n - |S| - 1)!}{n!} \big[f_{S \cup \{i\}}(x) - f_S(x) \big],$$

where $f_S(x)$ denotes the model's expected output when only the features in S are known. The formula can be read as a weighted average of feature i's *marginal contribution*. The bracketed term $f_{S\cup\{i\}}(x) - f_S(x)$ is the change in the prediction caused by adding feature i to a coalition S that does not already contain it. The sum ranges over *every* such coalition $S \subseteq N \setminus \{i\}$, so feature i is evaluated in all possible contexts of the other features, not in isolation. The combinatorial coefficient $|S|!\,(n - |S| - 1)!/n!$ is the fraction of feature orderings in which exactly the features in S precede i; weighting by it makes ϕ_i the average marginal contribution of i taken over all $n!$ orderings in which the features could be added, which is what gives the attribution its order-independence and its axiomatic guarantees.

From Shapley values to SHAP. SHAP (SHapley Additive exPlanations) [6] applies this construction to a machine-learning prediction by treating the input features as the "players" and the model output as the "payoff" to be distributed among them. The SHAP value ϕ_i of feature i is its average marginal contribution to the prediction over all orderings in which features are added, and the attributions satisfy *local accuracy*: together with a base value $\phi_0 = \mathbb{E}[f(x)]$, the expected prediction over a background distribution, they sum to the model's output for the instance, $f(x) = \phi_0 + \sum_{i=1}^{n} \phi_i$. A SHAP value is therefore read as the signed amount by which feature i moves the prediction above ($\phi_i > 0$) or below ($\phi_i < 0$) that baseline, and its magnitude expresses how strongly the feature influences the prediction for that instance.

Computing SHAP values exactly is expensive. The Shapley formula sums over all 2^n subsets of the n features, so an exact, model-agnostic computation grows *exponentially with the number of features* and quickly becomes intractable as the feature set scales. This motivates more efficient estimators. The model-agnostic KernelSHAP approximates the values by sampling feature coalitions, trading exactness for tractability, whereas for tree-based ensembles such as gradient-boosted decision trees and random forests TreeSHAP [7] exploits the tree structure to compute *exact* Shapley values in polynomial time, making it practical for the multi-model setting developed in this work.

Why gradient-boosted trees over neural networks. The predictive models throughout this thesis are gradient-boosted decision trees, LightGBM [28], XGBoost [29], and CatBoost [30], instead of neural networks. The choice is driven by both the modelling task and the explanation objective. The data are *tabular*: each instance is a modest-dimensional vector of simulation parameters mapped to a scalar KPI, with only a few thousand samples per domain (Section 6.1). In this regime gradient-boosted trees are consistently competitive with, and frequently superior to, deep neural networks, while requiring substantially less tuning to reach that performance [31]; the strong per-domain fits reported later in Section 7.1 bear this out. The decision also aligned with the modelling practice adopted within the wider industrial project

in which this thesis was conducted, where gradient-boosted trees were already the working choice for KPI prediction on this class of tabular telemetry, so the framework was developed against the model family it would be expected to explain in that setting.

More importantly, the choice serves the central object of study of this thesis, explanation *reliability*. Because TreeSHAP returns *exact* Shapley values for tree ensembles, the attribution stage introduces no sampling variance of its own. Since the Explanation Aggregation Metric developed in Section 5.4 quantifies precisely the stability and fidelity of attributions, an exact explainer ensures that the metric measures genuine differences in explanation quality between models and not the Monte-Carlo noise of an approximate explainer. Neural networks would restrict attribution to sampling-based estimators such as KernelSHAP or DeepSHAP, whose approximation error would be entangled with the very reliability signal the metric is designed to isolate, exactly the difficulty that arises for the black-box composite operators of Section 6.5. The aggregation framework itself is agnostic to the model family (Section 4), so neural predictors remain compatible in principle; gradient-boosted trees are adopted for the empirical evaluation because they allow explanation reliability to be studied without attribution-method noise as a confounding factor.

2.3 5G Network Architecture and Functional Domains

The 5G New Radio (NR) radio-access technology was introduced by 3GPP in Release 15 and extended in subsequent releases [19]. A 5G mobile network can be decomposed into a set of functional domains: the radio access network (NG-RAN) [19], the transport/backhaul, and the 5G Core [32]. Their *joint* behaviour determines end-to-end service quality. Figure 1 illustrates how a User Equipment (UE) request traverses the RAN, the Transport/Backhaul layer, and the 5G Core before reaching the application server, and how each domain contributes latency, capacity constraints, and availability risk to the overall service. Understanding this decomposition is essential to the present work because the explainability framework proposed in Section 5 aggregates per-domain SHAP attributions across exactly these three layers.

Radio Access Network (RAN). The RAN comprises one or more gNBs (next-generation base stations) and the radio interface connecting UEs to the network. Key RAN parameters include the channel bandwidth (MHz), the numerology index $\mu \in \{0, \ldots, 4\}$ which determines the sub-carrier spacing $\Delta f = 2^{\mu} \times 15\,\text{kHz}$ and thus the slot duration, the centre frequency, the cell radius, and UE mobility (speed in km/h). These parameters jointly govern the achievable spectral efficiency via the Shannon bound and the SINR distribution, and therefore impose a hard ceiling on application-layer throughput. Scheduling delay and HARQ retransmissions introduce stochastic latency components that are highly sensitive to channel quality and interference. In sparse or lightly loaded deployments the RAN is typically the dominant throughput bottleneck; its availability KPI reflects the radio link failure rate, which is highly variable under mobility and fading and therefore exhibits higher variance than the corresponding core metric under most operational regimes.

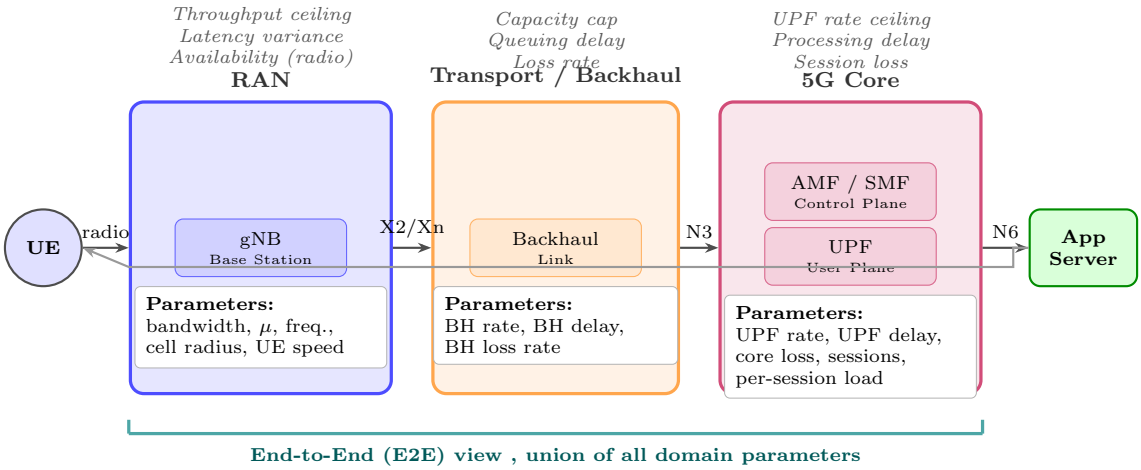


Figure 1: 5G network architecture decomposed into the three functional domains used throughout this thesis. User traffic (solid arrows) flows upward from the UE through the RAN, the Transport/Backhaul layer, and the 5G Core to the application server; the return path is shown in light grey. Each domain introduces distinct constraints on throughput, latency, and availability, labelled above the corresponding block. The E2E view (teal bracket) treats the union of all domain parameters as a single feature space, enabling cross-domain attribution.

Transport / Backhaul Network. The backhaul connects gNBs to the 5G Core over fibre, microwave, or satellite links. Its capacity (backhaul rate in Mbps), propagation delay (ms), and packet loss rate (%) collectively constrain the traffic that the radio interface can deliver to the core. When offered load approaches the backhaul rate, queues build, loss increases, and end-to-end latency grows sharply and non-linearly, a behaviour that motivates the inclusion of a separate transport KPI model instead of folding backhaul effects into the E2E model alone. Backhaul parameters are typically operator-configured, not dynamically negotiated, making them good candidates for counterfactual analysis: "what would E2E latency be if the backhaul delay were halved?"

5G Core (5GC). The 5GC separates the control plane, managed by the Access and Mobility Management Function (AMF) and Session Management Function (SMF), from the user plane, managed by the User Plane Function (UPF). From a KPI modelling perspective the relevant parameters are the UPF processing rate (Mbps), UPF internal delay (ms), core loss rate, number of simultaneous sessions, and per-session offered load. Under nominal conditions the core is high-throughput and near-deterministic in latency; however, UPF overload or session saturation causes abrupt degradation. This near-bimodal behaviour, nominal performance or severe degradation, means that core availability is near-deterministic under most operating points but can collapse discontinuously, a statistical property that is qualitatively different from both RAN and transport KPI distributions and justifies separate per-domain modelling.

End-to-End (E2E) View. The E2E view integrates the parameters of all three domains into a single feature space and models the KPIs as experienced at the application layer. E2E KPIs reflect the minimum bottleneck across all domains for throughput (i.e. $\text{Thr}_{\text{E2E}} \leq \min(\text{Thr}_{\text{RAN}}, \text{Thr}_{\text{BH}}, \text{Thr}_{\text{UPF}})$) and the sum of all indi-

vidual delays for latency. In the proposed framework the E2E view is modelled by its own trained predictors and serves as a *benchmark*: it enables direct end-to-end attribution that may reveal interaction effects between parameters from different domains, and provides a reference against which the composed (QoS-algebra) explanations are compared. It is not treated as a fourth aggregation domain: the cross-KPI and cross-domain aggregations operate only over the three functional domains (RAN, Transport, Core).

2.4 QoS Key Performance Indicators

Three Key Performance Indicators (KPIs) are used throughout this thesis to characterise service quality at both the per-domain and end-to-end levels. They are chosen to reflect three orthogonal dimensions of network service quality typically expressed in operator intents and 3GPP QoS profiles: capacity, responsiveness, and reliability.

- **Throughput (Mbps).** The application-layer goodput delivered to the user, averaged over the observation period. End-to-end throughput is bounded above by the minimum sustainable rate across the RAN, transport, and core domains: it cannot exceed the spectral efficiency achievable on the radio link, the capacity of the backhaul path, or the forwarding rate of the User Plane Function (UPF). Its empirical distribution is approximately Gaussian in the uncongested regime and becomes right-skewed as any one domain approaches saturation.
- **Latency (ms).** The mean one-way packet delay from the User Equipment (UE) to the application server. End-to-end latency decomposes additively into contributions from each domain: RAN scheduling and transmission delay, backhaul propagation and queuing delay, UPF queuing delay, and core processing delay. Latency is highly load-sensitive and typically exhibits bimodal behaviour, with a low-congestion regime in which delay is dominated by fixed propagation and processing components, and a near-saturation regime in which queuing delay grows rapidly and dominates the total.
- **Availability (%).** The fraction of time during which the service is operational, commonly operationalised as the proportion of observation intervals in which the packet-delivery ratio exceeds a fixed threshold. Each domain contributes a distinct failure mode: in the RAN, availability reflects radio link failure rate and is sensitive to UE mobility and channel quality; in the transport domain it reflects loss-induced session drops; and in the core it reflects session saturation and control-plane failures. The statistical regimes across domains differ substantially: core availability is near-deterministic under typical operating conditions, whereas RAN availability exhibits high variance driven by stochastic channel effects.

The statistical properties of these KPIs vary substantially across domains, and this heterogeneity has consequences for any system that attempts to predict or explain end-to-end service quality from per-domain signals. The three domains exhibit distinct statistical regimes:

- **Core.** Availability is near-deterministic under typical operating conditions. Failures are rare, highly correlated with a small number of control-plane events,

and produce a strongly imbalanced distribution dominated by the "available" class.

- **RAN.** Availability exhibits high variance driven by the volatile radio channel, user mobility, and interference. The resulting distribution has much heavier tails and a far richer dependence on input features such as Signal-to-Interference-plus-Noise Ratio (SINR), Channel Quality Indicator (CQI), and offered load.
- **Transport.** Availability sits between these extremes, with behaviour governed by a mixture of deterministic link provisioning and stochastic loss events.

Latency and throughput display analogous cross-domain asymmetries: RAN-side measurements are noisy and regime-dependent, whereas core-side measurements are comparatively stable and tightly bounded.

Several practical risks follow from this heterogeneity:

1. **Variance dominance in pooled training.** A model trained on pooled multi-domain data tends to be dominated by the highest-variance domain, because the loss function is driven by the residuals where the signal is noisiest. Low-variance domains then contribute little to the fitted parameters, even when they are operationally critical.
2. **Non-comparable attributions across domains.** Feature-importance scores computed on such a pooled model are not directly comparable across domains: a feature in a near-deterministic domain may receive a small attribution simply because the target barely varies, not because the feature is operationally unimportant.
3. **Class imbalance on rare events.** Class imbalance in availability, particularly in the core, can cause naive classifiers to achieve high accuracy by predicting the majority class while failing precisely on the rare events that intent-assurance systems most need to detect.
4. **Heterogeneous calibration.** The differing noise regimes mean that the calibration of predicted probabilities, and hence the validity of any threshold-based intent check built on top of them, is unlikely to be uniform across domains unless calibration is performed per-domain.

These concerns motivate treating per-domain predictions and their explanations as objects with distinct statistical characters that must be combined deliberately, not as exchangeable quantities that can be averaged or concatenated without further consideration.

3 Related Work

3.1 Explanation Quality Evaluation

The question of what constitutes a good explanation has received significant attention in the XAI literature, with fidelity and robustness emerging as two foundational criteria. Yeh et al. [13] formalise *infidelity* as the mean squared difference between an explanation's predicted output change under a feature perturbation and the model's actual output change, and show that explanations optimised to reduce it are more faithful to the underlying model. Their formulation is attractive because it reduces a qualitative notion of faithfulness to a measurable quantity that can be computed for any black-box model, which makes it a natural starting point whenever explanation quality must be reported alongside attributions themselves. On this basis, the fidelity component of the Explanation Aggregation Metric used in this thesis follows the (in)fidelity formulation of [13], with feature masking substituted for signed perturbation so that the evaluation remains compatible with the magnitude-based attribution semantics employed here.

Robustness to input variation is a complementary concern. Alvarez-Melis and Jaakkola [14] show that popular attribution methods, including LIME and SHAP, can be highly sensitive to small input perturbations: nearly identical inputs may yield substantially different explanations. To capture this instability formally, they propose a local Lipschitz-based continuity criterion that bounds the change in an explanation, within a neighbourhood of a given input, by the magnitude of the change in the input. Their analysis is conducted on individual models in isolation, leaving open the question of how attribution stability should be quantified when explanations from several models are combined, as is the case in multi-domain settings. The Stability component of the Explanation Aggregation Metric (Section 5.4) is directly inspired by this view: attribution variance under Gaussian perturbations is measured and inverted into a $[0, 1]$ reliability score, which can then be used as a per-model weight when attributions are aggregated.

A quantitative technique for scoring the strength of a post-hoc explanation, the Level of Strength of an Explanation (LSE), is proposed by Bello et al. [12]. Motivated by the absence of ground truth for what constitutes a correct explanation, it borrows the semantics of the likelihood ratio used to weigh forensic evidence, quantifying the extent to which an explanation supports the class predicted by a model. Their measure is developed for the single-model classification setting and yields a single scalar used to rank explanation methods. This thesis shares their goal of reducing explanation quality to a computable score, but targets a regression and multi-domain context and combines several axes, fidelity, stability, and consistency, with agreement *between* per-domain explanations itself treated as a quality signal.

Rieger and Hansen [33] argue that disagreement between explanation methods, the so-called *disagreement problem* [34, 35], can be mitigated by aggregating across methods. Their setting is one in which a single model is explained by several different attribution techniques, and aggregation serves to reconcile methodological disagreement. The setting addressed in this thesis is structurally different: the same attribution method is applied to several different models, each trained on a different network domain, and aggregation serves to reconcile *domain-induced* disagreement rather than methodological disagreement. Mothilal et al. [25] further connect feature attribution and counterfactual explanation frameworks, suggesting

that complementary XAI approaches are best viewed as facets of a single explanatory goal instead of competing alternatives, a perspective that supports the choice in this thesis to treat per-domain attributions as complementary views of a shared end-to-end quantity.

3.2 Explanation Aggregation Across Models

The problem of combining explanations from multiple models has been studied from several angles, though no prior work addresses the multi-domain 5G network setting. Bhatt et al. [36] propose methods for evaluating and aggregating feature-based explanations, characterising an explanation along three axes: sensitivity, faithfulness, and complexity. In their setting several attribution methods applied to a single model are combined into one aggregate explanation, and they show that naive averaging can inflate explanation complexity, motivating aggregation that optimises a chosen quality criterion. Their work establishes the principle that aggregation should be quality-aware, but it operates over multiple explanation methods for a single model rather than over multiple models. The reliability-based weighting mechanism developed in this thesis carries that principle into the multi-model setting, using fidelity and stability as the operational quality signals whose combination is given a closed form.

Vascotto et al. [37] similarly propose aggregating explanations from multiple independently trained models to improve trustworthiness, using inter-model consensus as a quality signal. A natural assumption in their setting is that the constituent models are interchangeable: trained on the same task with the same input space, they are expected to agree, and disagreement is taken as evidence of unreliability. This assumption does not transfer to the multi-domain setting addressed here, where models are trained on different tasks and different feature spaces and disagreement may reflect genuine domain heterogeneity, not unreliability. The framework in this thesis therefore introduces L1-normalisation prior to aggregation, restoring scale comparability without forcing the spurious agreement that pooled training would impose.

Chatterjee et al. [38] propose a multi-criteria rank-based framework that synthesises explanations according to multiple evaluation criteria using Borda count and related rank aggregation methods. Rank aggregation is well suited to their setting, in which the question is which of several competing explanation methods ranks features most consistently for a single model.

The setting in this thesis is dual to theirs: a single explanation method is applied to multiple models, and the quantity of interest is the magnitude of attributions rather than their relative ranking, since intent-assurance decisions require a numerical signal that can be thresholded. Magnitude-preserving aggregation is therefore preferred to rank aggregation, a distinction motivated by the interpretability uniformity requirements of network operations.

3.3 XAI for 5G and O-RAN

The intersection of XAI and 5G/6G network management has attracted growing research interest, driven by the proliferation of machine learning components in Open RAN (O-RAN) architectures. Brik et al. [22] provide a comprehensive survey on

XAI in 6G O-RAN, covering architectural integration points, use cases, and practical challenges including the explainability-performance trade-off, standardisation, and the interoperability and privacy of XAI models across multi-vendor O-RAN deployments. As one direction, they envisage composing vendor-specific local XAI models into a global, interoperable model at the Non-Real-Time RIC, but treat this as a forward-looking architectural concern and offer neither a concrete mechanism for combining the resulting explanations nor a quantitative methodology for assessing whether such combination preserves explanation quality. This thesis addresses that gap, contributing both a weighted aggregation mechanism and a quantitative quality metric that can be applied to it.

Chari et al. [26] develop an Explanation Ontology that provides a semantic representation for user-centred explanations, covering explanation types, user models, and explanation goals. Their contribution is at the representational layer: it gives a vocabulary for *describing* explanations but is deliberately agnostic about how the underlying numerical attributions are produced or combined.

The numerical aggregation framework proposed here operates at a lower layer and is in principle compatible with their ontology: the outputs of the framework can be categorised using their explanation-type vocabulary when integrated into a user-facing system.

3.4 Intent-Based Explainability Orchestration

The most closely related work to the cross-domain aggregation component of this thesis is that of Bilgin et al. [2], who propose an intent-based end-to-end XAI orchestration framework for AI-native networks. Their framework maps network intents to explanation requirements and orchestrates XAI queries across distributed AI components to produce intent-aligned explanations. The emphasis of their work is on the *control plane* of explanation: deciding which component to query, when, and on whose behalf, given a stated intent. Quality of the resulting explanations, and how explanations from different components should be reconciled when they overlap, is not the focus of their framework.

The framework developed in this thesis can be viewed as addressing the data plane of the same problem: given that several components have already been queried, how should their attributions be combined into a single intent-aligned explanation, and how should the quality of that combined explanation be measured? The two perspectives are complementary, and the principal contributions of this thesis relative to [2] are a quality-aware weighting mechanism, a quantitative evaluation metric, and an empirical evaluation on realistic multi-domain 5G data.

3.5 Positioning of This Work

Table 2 summarises the most relevant prior works along five dimensions: multi-model aggregation, quality-informed weighting, formal aggregation, applicability to a 5G or network domain, and an intent layer. To the best of the authors' knowledge, no prior work simultaneously addresses all five dimensions in a unified, quantitatively evaluated framework. The principal gap in the literature is the absence of a principled, reliability-aware attribution aggregation mechanism validated on realistic multi-domain 5G data with a connection to intent-based network management.

Table 2: Comparison of related works. ✓= addressed, ∼ = partially addressed, – = not addressed.

Work	Multi-model aggregation	Quality weighting	Aggregation formalism	5G/network domain	Intent layer
Bhatt et al. [36]	✓	∼	✓	–	–
Rieger & Hansen [33]	✓	∼	✓	–	–
Vascotto et al. [37]	✓	∼	✓	–	–
Chatterjee et al. [38]	–	✓	✓	–	–
Brik et al. [22]	–	–	–	✓	∼
Bilgin et al. [2]	✓	–	∼	✓	✓
Bello et al. [12]	–	✓	–	–	–
This thesis	✓	✓	✓	✓	✓

4 Problem Formulation

4.1 System Overview

Consider a 5G network managed by an AI-driven control plane that relies on a collection of predictive models. The models are organised by domain $d \in \mathcal{D} = \{\text{RAN, Transport, Core}\}$ and by KPI $\kappa \in \mathcal{K} = \{\text{throughput, latency, availability}\}$. A further End-to-End (E2E) model per KPI is trained on the union feature space; it is used as a *benchmark* against which the composed cross-domain explanations are compared and is *not* a member of the aggregation set $\mathcal{D}$. For each pair (d, κ), a dedicated regression model $M_{d,\kappa}$ has been trained on domain-specific features:

$$M_{d,\kappa} : \mathbb{R}^{p_d} \longrightarrow \mathbb{R},$$

where p_d is the number of features in domain d. The feature set depends on the domain, not the KPI: all three per-KPI models of a domain share the same feature set $\mathcal{F}_d \subseteq \mathcal{F}$ with $|\mathcal{F}_d| = p_d$, where $\mathcal{F} = \bigcup_d \mathcal{F}_d$ is the global feature union used only when comparing across domains.

For each model and a given input x, a SHAP explainer $\Phi_{d,\kappa}$ produces a feature attribution vector:

$$\phi^{(d,\kappa)} = \Phi_{d,\kappa}(x) \in \mathbb{R}^{p_d},$$

where $\phi_j^{(d,\kappa)}$ quantifies the contribution of feature j to the model's prediction at x.

4.2 Problem Statement

P1: Cross-KPI attribution incomparability. Feature-attribution methods inherit the units of the model's output: attributions explain a prediction by decomposing it additively (or approximately so) into per-feature contributions, and those contributions are necessarily expressed in the same units as the prediction itself. A throughput-model attribution is therefore in Mbps, a latency-model attribution in milliseconds. Even when both refer to the same feature j, the quantities $\phi_j^{(\text{RAN,tput})} = 2.3\,\text{Mbps}$ and $\phi_j^{(\text{Core,lat})} = 0.4\,\text{ms}$ are dimensionally incommensurable: they cannot be summed or averaged in any physically meaningful way, regardless of how their numerical magnitudes happen to compare.

P2: Unequal explanation reliability. Not all $M_{d,\kappa}$ models generate equally reliable explanations. A model that is overfitted, trained on noisy data, or whose prediction surface changes rapidly across the input space (e.g., the latency model under regime transitions) will produce SHAP vectors that are less stable and less faithful than those from a well-calibrated model. Aggregating without accounting for this heterogeneity biases the result towards unreliable sources.

P3: Absence of a consolidated feature importance representation. Network operators and engineers require an interpretable feature-importance summary that answers "which parameters matter most for network quality": resolved *per domain* across that domain's KPIs, and, *across domains*, tied together through the physical way in which each KPI composes end to end. No such representation is produced by any single model, and no principled aggregation method currently exists for the multi-domain 5G context.

4.3 Formal Problem Definition

Input: A set of $|\mathcal{D}| \times |\mathcal{K}|$ trained models $\{M_{d,\kappa}\}_{d\in\mathcal{D},\,\kappa\in\mathcal{K}}$, each with a corresponding SHAP explainer that produces attribution vectors $\phi^{(d,\kappa)} \in \mathbb{R}^{p_d}$, and a test set $\mathcal{X}_{\text{test}}$.

Goal: Along the *intra-domain, cross-KPI* axis, compute for each domain d a *domain-level attribution profile* $\Phi^{(d)} \in \mathbb{R}^{p_d}$ that summarises which parameters drive that domain's behaviour across its KPIs, such that:

1. $\Phi^{(d)}_j \in [0,1]$ for all $j \in \mathcal{F}_d$, and $\sum_j \Phi^{(d)}_j \leq 1$ (normalised, non-negative).
2. $\Phi^{(d)}$ is a convex combination of that domain's per-KPI attribution vectors $\{\phi^{(d,\kappa)}\}_{\kappa\in\mathcal{K}}$, weighted according to the explanation quality of each per-KPI model. This is well defined without re-indexing because the per-KPI models share the domain feature set $\mathcal{F}_d$.
3. The weighting scheme down-weights attributions from models whose explanations are unreliable (low fidelity, low stability, or low consistency) and up-weights those from models with higher quality explanations.
4. The dominant drivers identified by $\Phi^{(d)}$ are robust to the choice of weighting scheme: reliability weighting reorders only lower-confidence, middle-ranked features and does not displace the top-ranked drivers relative to uniform averaging.

Along the complementary *cross-domain, same-KPI* axis, the per-domain attributions of a fixed KPI are composed into an end-to-end explanation through that KPI's physical QoS operator (sum, minimum, or product; Section 5.6) over the union feature space $\mathcal{F}$, *not* by reliability weighting. The two axes are treated distinctly, and the E2E model serves only as the benchmark against which the cross-domain compositions are compared.

4.4 Scope and Assumptions

The framework developed in this thesis is formulated for *attribution-based* explanation methods, that is, methods that assign to each input feature a scalar importance

value for a given prediction. Both the weighted aggregation mechanism and the Explanation Aggregation Metric operate on attribution vectors and are, in principle, agnostic to the specific method used to produce them. Within this general setting, the following scope restrictions and assumptions apply:

1. **SHAP as the attribution method.** SHAP [6] is adopted as the concrete attribution method throughout this thesis. The choice is motivated by two properties that the aggregation framework relies on: *local accuracy*, which guarantees that per-feature attributions sum to the model output offset from a baseline and thus admit a direct quantitative interpretation, and the *additivity* of Shapley values [27], which makes cross-model combination of attributions well-defined. Other attribution methods that satisfy these properties could be substituted without altering the framework; methods that do not, such as raw gradients or purely rank-based attributions, would require adaptation and are outside the present scope.

2. **Tree ensemble models.** All domain models are gradient-boosted tree ensembles (LightGBM, XGBoost, or CatBoost). This choice is instrumental rather than fundamental: it enables the use of TreeSHAP [7], which computes exact SHAP values in polynomial time and removes sampling variance from the attribution stage. Neural networks and other model families are compatible with the framework via KernelSHAP or DeepSHAP but are outside the scope of the empirical evaluation.

3. **Magnitude-based aggregation.** Aggregation is performed on absolute (magnitude) SHAP values. Signed values preserve the direction of effect but are not directly comparable across models predicting different quantities, since the sign convention is target-specific. Magnitude aggregation yields a universal "importance" measure that is comparable across domains and is sufficient for the intent-assurance use case targeted here.

4. **Simulation data.** All training and evaluation data are generated from the ns-3/NR simulator. Real-world measurement data collection is outside the scope of this thesis.

5. **Static quality evaluation.** Quality scores (EAM) are computed once on the test set and used as fixed weights during aggregation. Dynamic, per-instance quality computation is left as future work.

6. **Uniform base weights.** No prior domain or KPI priority weights are imposed ($\alpha_\kappa \equiv 1$). Extensions with non-uniform base weights, for example to encode operator preferences or service-level priorities, are discussed in Section 8.2

5 Methodology

This section describes the complete methodology: simulation data generation, feature engineering, model selection and training, feature attribution computation, Aggregation scoring, the aggregation algorithm, and the QoS algebra extension.

5.1 Data Generation via ns-3 Simulation

All training and evaluation data are generated using the ns-3 network simulator [39] with the 5G NR module [40]. A parameter sweep approach is used: for each configuration, a simulation run produces a single data point containing the input parameters and the resulting KPI measurements.

5.1.1 Simulation Setup

The simulated network consists of a single cell with a gNB and a variable number of UEs, a backhaul link connecting the gNB to the core, and a UPF handling session traffic. The simulation duration is fixed at 10 s of application traffic, preceded by a connection establishment phase. UDP traffic with configurable packet size and inter-packet interval is used.

5.1.2 Parameter Sweep

Parameters are swept over their respective ranges in a grid-and-random hybrid fashion, described in Table 3. The sweep covers ∼2,900 configurations per domain, with ranges chosen to cover typical 5G operating conditions as well as edge cases (overloaded backhaul, near-saturation UPF, high UE density).

Table 3: Key simulation parameters and their swept ranges.

Domain	Parameter	Range	Description
RAN / E2E	`num_ue`	1–50	Number of active UEs
	`bandwidth_hz`	10–100 MHz	Channel bandwidth
	`numerology`	0–3	Sub-carrier spacing index
	`center_freq_hz`	3.5–28 GHz	Centre frequency
	`cell_radius_m`	50–500 m	Cell radius
	`ue_speed_mps`	0–30 m/s	UE velocity
Transport / E2E	`backhaul_rate_mbps`	10–1000 Mbps	Backhaul capacity
	`backhaul_delay_ms`	1–50 ms	Backhaul propagation delay
	`backhaul_loss_pct`	0–5%	Backhaul loss rate
Core / E2E	`upf_rate_mbps`	10–1000 Mbps	UPF forwarding rate
	`upf_delay_ms`	0.1–20 ms	UPF processing delay
	`num_sessions`	1–100	Number of sessions
	`core_loss_pct`	0–5%	Core loss rate

5.1.3 Feature Engineering

Raw simulation parameters are augmented with derived features that capture physically meaningful interactions. Key engineered features include:

- `offered_load_mbps`: total offered traffic load (= num_ue × load_per_ue_mbps).
- `bh_util_ratio`, `upf_util_ratio`, `ran_util_ratio`: per-domain utilisation ratios (offered load divided by domain capacity), serving as proxies for congestion in the Transport, Core, and RAN domains respectively.
- `eff_bh_rate_mbps`, `eff_upf_rate_mbps`: effective bottleneck rates (= min(offered load, domain capacity)).
- Quadratic overload indicators: `bh_util_sq`, `upf_util_sq`, enabling the model to capture non-linear saturation behaviour.
- Queue-factor features (`bh_queue_factor`, `upf_queue_factor`, `cpu_queue_factor`, `ran_queue_factor`): derived proxies for queuing delay growth under load.
- Log-transforms of skewed features: `log_num_ue`, `log_bandwidth_mhz`, `log_ue_density`, `log_doppler_hz`, `log_bh_rate`, `log_link_distance`, `log_upf_rate`, `log_upf_proc_ms`.

The E2E feature set is the union of all per-domain features, with domain-prefixed copies of shared parameters (e.g. `bh_num_ue`, `core_offered_load_mbps`) retained to allow the E2E model to learn domain-specific effects independently. The complete feature sets for all four domains, with final counts of 22 (RAN), 21 (Transport), 20 (Core), and 63 (E2E) excluding the identifier column `run_id`, are listed in Appendix A (Table 6).

5.1.4 Train/Validation/Test Split

Each domain dataset is partitioned into three disjoint subsets with distinct roles. The *training* set (70%) fits the model parameters; the *validation* set (15%) drives model selection and early stopping, so that the choice of model family and the point at which training halts are decided without ever consulting the test data; and the *test* set (15%) is held out untouched, used only for the final performance estimate and for computing all SHAP attributions and EAM scores. Separating the data that *tunes* a model from the data that *judges* it prevents the optimistic bias that arises when a single set serves both purposes, so the reported R^2 values and the downstream reliability scores reflect genuine generalisation.

The split is performed *chronologically* (by sweep order). Neighbouring configurations in a parameter sweep are highly similar, so a random split would place near-identical points on both sides of the partition and leak information from the test set into training; a chronological cut avoids this and additionally mirrors the deployment setting, in which a model trained on past configurations must generalise to configurations it has not yet seen. Finally, features are standardised (zero mean, unit variance) using statistics computed on the training set only, so that no information from the validation or test sets enters the preprocessing stage.

5.2 Model Training and Selection

For each domain-KPI pair (d, κ), three gradient-boosted tree models are trained: LightGBM [28], XGBoost [29], and CatBoost [30]. The best model is selected based

on validation-set Root Mean Squared Error (RMSE). All models are trained with early stopping (50 rounds patience) monitored on the validation set to prevent overfitting.

Performance is evaluated on the held-out test set using the root mean squared error (RMSE) and the coefficient of determination (R^2):

$$R^2 = 1 - \frac{\sum_i (y_i - \hat{y}_i)^2}{\sum_i (y_i - \bar{y})^2},$$

where y_i is the true target value for sample i, $\hat{y}_i$ is the model's predicted value, and $\bar{y}$ is the mean of the true target values over the test set.

5.3 Feature Attribution Computation

Feature attributions are computed using TreeSHAP [7], applied to each best-selected model on the test set. The SHAP library's `TreeExplainer` is used directly on the native booster for all three model families (LightGBM, XGBoost, and CatBoost), which it supports through the same TreeSHAP algorithm.

The output for model k is an attribution matrix $\Phi^{(k)} \in \mathbb{R}^{n \times p_k}$, where n is the test set size and $p_k = |\mathcal{F}_k|$ is the number of features. Each row $\phi_i^{(k)}$ is the attribution vector for test instance i, decomposing the model's prediction into per-feature contributions.

For composite models (Section 5.6), KernelSHAP is used, since the composites are not themselves tree models; its background distribution is a 100-instance random sample of the union test set, and each explained instance is attributed using 500 coalition samples (Section 5.6.4).

5.4 Aggregation Metric

5.4.1 Overview

The Explanation Aggregation Metric score $\text{EAM}_k \in [0, 1]$ for model k is defined as:

$$\text{EAM}_k = \frac{F_k + S'_k + C_k}{3}, \tag{1}$$

where F_k (Fidelity), S'_k (Stability), and C_k (Consistency) are each bounded in $[0, 1]$. Higher values indicate higher explanation quality.

5.4.2 Fidelity

Fidelity measures how faithfully the attribution vector captures the model's dependence on its top features. For each test instance x:

1. Identify the top-3 features by absolute attribution value.
2. Construct a masked input $x^{(\text{mask})}$ by replacing those features with their training-set column means.
3. Compute the prediction change: $\delta_i = |f(x_i) - f(x_i^{(\text{mask})})|$.

Fidelity is then:

$$F_k = \text{clip}\left(\frac{\overline{\delta}}{\sigma_f},\ 0,\ 1\right), \tag{2}$$

where $\overline{\delta} = \frac{1}{n}\sum_{i=1}^{n}\delta_i$ is the mean prediction drop and σ_f is the standard deviation of model predictions on the test set. Division by σ_f makes F_k scale-invariant across KPIs with different output ranges (Mbps vs. ms vs. %).

5.4.3 Stability

Stability measures the variance of attribution values under Gaussian input perturbations [14]. For $r = 5$ perturbation rounds:

1. Sample $\varepsilon \sim \mathcal{N}(0, \delta \cdot \sigma_{\text{feat}})$, where $\delta = 0.02$ and σ_{feat} is the per-feature standard deviation on the test set.
2. Compute attribution values $\phi^{(k)}(x_i + \varepsilon^{(t)})$ for each round t.
3. Compute the variance of attribution values across rounds for each (instance, feature) pair.

The mean variance across all (instance, feature) pairs is $\bar{V}_k$, and Stability is:

$$S'_k = \frac{1}{1 + \bar{V}_k}. \tag{3}$$

This is an inverse-variance score: $S'_k = 1$ when SHAP values are perfectly stable ($\bar{V}_k = 0$) and approaches 0 as variance grows.

5.4.4 Consistency

Consistency measures the coherence of attribution vectors across test instances. Every instance in a domain is described by the *same* set of features and differs only in the feature *values*, so Consistency asks whether the model attributes importance to those shared features in a comparable way as their values change, instead of reweighting them differently from one input to the next. A model with consistent explanations applies similar feature weightings across diverse inputs. Consistency is defined as the mean pairwise cosine similarity of attribution vectors across the test set, rescaled to $[0, 1]$:

$$C_k = \text{clip}\left(\frac{\overline{\cos(\phi^{(k)}(x^{(i)}),\ \phi^{(k)}(x^{(j)}))} + 1}{2},\ 0,\ 1\right), \tag{4}$$

where the mean is taken over all pairs (i, j) with $i \neq j$ in the test set, and cosine similarity is computed on the signed SHAP vectors. The $(\cdot + 1)/2$ rescaling maps $[-1, 1]$ to $[0, 1]$.

High C_k indicates that the model consistently assigns importance to the same features regardless of the specific input. Low C_k may indicate that the model's explanation relies on different features in different operating regimes, which can be a genuine property of the model (not a flaw) but reduces the trustworthiness of any single explanation as a global characterisation.

5.5 Aggregation Algorithm

The aggregation algorithm combines per-model SHAP attribution vectors into a single global feature-importance ranking. The complete procedure is given in Algorithm 1; in outline it (i) reduces each model's per-sample attributions to a mean vector and lifts it onto the common (union) feature space, (ii) L1-normalises that vector so every model contributes one unit of explanatory mass, (iii) scores each model's explanation reliability with the EAM of Section 5.4, and (iv) sums the normalised vectors in proportion to those reliability weights. The individual steps are detailed beneath the algorithm.

This EAM-weighted procedure applies to the *cross-KPI* (intra-domain) axis only. A single functional domain is fixed, and its $K = 3$ per-KPI models (throughput, latency, and availability) are aggregated into one domain-level profile, so the index k ranges over exactly these three models. This is the only place EAM weighting is used. The three functional domains (RAN, Transport, Core) are each aggregated this way; the E2E model is a benchmark and is never included.

The complementary aggregation (combining the three per-domain models for a *fixed* KPI into a single end-to-end estimate) is *not* performed by EAM-weighted averaging. Per-domain KPI predictions compose according to the physical semantics of the KPI (latency sums across segments, throughput is bounded by the minimum-capacity domain, availability multiplies under independent failure), not according to explanation-reliability weights. This cross-domain composition is therefore carried out by the QoS-algebra operators defined in Section 5.6, over the same three functional domains, with E2E again treated as a benchmark rather than a composed domain.

The shape of the algorithm follows directly from the ways in which the per-model explanations are *not* comparable; each design choice addresses one such source of incomparability, not a choice included for its own sake.

- **L1 normalisation (differing scales).** Raw SHAP attributions are expressed in the units of each model's prediction, so a model predicting throughput in Mbps and one predicting availability as a probability produce attributions on entirely different scales; summing them directly would let the largest-scale model dominate the aggregate for reasons unrelated to feature importance. Reducing every model to the same total mass of one means that what is combined is each feature's *share* of a model's importance, not its raw magnitude.

- **Absolute values (sign cancellation).** Importance is a question of how strongly a feature moves the prediction, not in which direction; keeping the sign would let positive and negative attributions cancel and understate genuinely influential features. The direction of effect is instead preserved separately in the signed vectors.

- **Alignment onto a common feature space.** The three per-KPI models aggregated within a domain are all trained on that domain's feature set, so their attribution vectors are already defined over the same features and are directly addable without re-indexing. The zero-padding onto the union feature space $\mathcal{F}$ shown in Algorithm 1 is stated for generality, so that the same procedure applies unchanged should the framework later combine models with differing

feature sets; in the intra-domain, cross-KPI setting used here it reduces to the identity.

- **EAM weighting (differing reliability).** Most important for this thesis, the explanations are not equally trustworthy: some models yield stable, faithful, and consistent attributions while others do not. A uniform average would treat an unreliable explanation as equal evidence to a reliable one; weighting each model by its EAM score instead lets more reliable explanations speak louder while still *retaining*, rather than discarding, the partial signal from weaker ones.
- **Convex combination (interpretability).** Using non-negative weights that sum to one keeps the final profile on the same $[0, 1]$ scale as the L1-normalised per-model shares it combines, so the aggregate is itself a normalised distribution of importance mass whose entries remain bounded and directly comparable across features instead of growing with the number of models averaged.

Algorithm 1 Explanation Aggregation

Require: Models of K concepts $\{M_k\}_{k=1}^{K}$; Explanation $\{\phi_{1..k} \in \mathbb{R}^{n\times|\mathcal{F}|}\}$; Explanation metrics $Q_k(\phi)$; feature sets $\{\mathcal{F}_k\}$
Ensure: Global importance vector $\Phi \in \mathbb{R}^{|\mathcal{F}|}$
1: **for** each model $k = 1, \ldots, K$ **do**
2: $Z_k^{(i)} \leftarrow \sum_{j\in\mathcal{F}_k} \left|\phi_{k,j}^{(i)}\right|$ for each sample i
3: $\tilde{\phi}_{k,j}^{(i)} \leftarrow \dfrac{\left|\phi_{k,j}^{(i)}\right|}{Z_k^{(i)} + \varepsilon}$ for all $j \in \mathcal{F}_k$, each sample i ▷ Per-sample L1-normalised absolute attribution
4: $\tilde{\varphi}_k \leftarrow \dfrac{1}{n}\sum_{i=1}^{n} \tilde{\phi}_k^{(i)}$ ▷ Mean normalised vector over n samples ($\sum_j \tilde{\varphi}_{k,j} = 1$)
5: Zero-pad $\tilde{\varphi}_k$ to $\mathcal{F}$: $\tilde{\varphi}_{k,j} \leftarrow 0$ for all $j \in \mathcal{F} \setminus \mathcal{F}_k$
6: $F_k \leftarrow \mathrm{clip}\Big(\overline{|\,f(x) - f(x_{\mathrm{mask}})\,|}\Big/\,\sigma_f,\ 0,\ 1\Big)$ ▷ Fidelity (Q_F): normalised mean prediction drop under top-3 masking
7: $S'_k \leftarrow \dfrac{1}{1 + \mathbb{E}[\mathrm{Var}(\phi(x+\varepsilon))]}$ ▷ Stability ($Q_{S'}$): inverse variance under small perturbations ε
8: $C_k \leftarrow \mathrm{clip}\left(\dfrac{\overline{\cos(\phi_k(x_i),\,\phi_k(x_j))} + 1}{2},\ 0,\ 1\right)$ ▷ Consistency (Q_C): mean pairwise cosine similarity mapped to $[0, 1]$
9: $\mathrm{EAM}_k \leftarrow \dfrac{F_k + S'_k + C_k}{3}$ ▷ EAM score per model
10: **end for**
11: $w_k \leftarrow \dfrac{\mathrm{EAM}_k}{\sum_{m=1}^{K} \mathrm{EAM}_m}$ for all $k \in K$ ▷ Normalised Aggregation weights
12: $\Phi_j \leftarrow \sum_{k=1}^{K} w_k\, \tilde{\varphi}_{k,j}$ for all $j \in \mathcal{F}$ ▷ Aggregated global importance
13: **return** Φ

The steps of Algorithm 1 are as follows.

1. **Per-sample L1 normalisation.** For each model k and each test instance i, the raw SHAP vector is normalised on its absolute values,

$$\tilde{\phi}_{i,j}^{(k)} = \frac{|\phi_{i,j}^{(k)}|}{\sum_{m\in\mathcal{F}_k} |\phi_{i,m}^{(k)}|}, \tag{5}$$

so that $\sum_{j\in\mathcal{F}_k} \tilde{\phi}_{i,j}^{(k)} = 1$ and each instance contributes exactly one unit of explanatory mass regardless of the scale of the model's prediction. Taking absolute values *per instance*, before any averaging, prevents positive and negative attributions from cancelling across instances. If an instance's total mass $Z_i^{(k)} = \sum_{m\in\mathcal{F}_k} |\phi_{i,m}^{(k)}|$ is zero it carries no attribution and is floored by a small ε.

2. **Mean attribution vector.** The per-sample normalised vectors are averaged over the n test instances into a single mean vector $\tilde{\phi}^{(k)} = \frac{1}{n}\sum_{i=1}^{n} \tilde{\phi}_i^{(k)}$ (written $\tilde{\varphi}_k$ in Algorithm 1), which again sums to one, so that the aggregation operates on one representative normalised explanation per model instead of on individual predictions.

3. **Alignment.** Each mean vector is lifted onto the union feature space $\mathcal{F} = \bigcup_k \mathcal{F}_k$ by zero-padding the features a model does not observe ($\tilde{\phi}_j^{(k)} = 0$ for $j \in \mathcal{F} \setminus \mathcal{F}_k$), so that all models are expressed over the same coordinates and can be summed.

4. **Explanation-quality scoring.** Each model is scored by its EAM (Section 5.4), the equally weighted mean of the Fidelity, Stability, and Consistency components, $\mathrm{EAM}_k = (F_k + S'_k + C_k)/3 \in [0, 1]$. This is the reliability signal that determines how strongly the model contributes to the aggregate.

5. **Weight derivation.** The EAM scores are normalised across the K models of the domain,

$$w_k = \frac{\mathrm{EAM}_k}{\sum_{m=1}^{K} \mathrm{EAM}_m}, \tag{6}$$

so that $w_k \geq 0$ and $\sum_k w_k = 1$; models with higher explanation quality receive proportionally larger weights. If every model has $\mathrm{EAM}_k = 0$, a uniform fallback $w_k = 1/K$ is applied so that the aggregation remains well-defined.

6. **Weighted aggregation.** The global attribution for feature j is the reliability-weighted sum of the normalised vectors,

$$\Phi_j = \sum_{k=1}^{K} w_k \, \tilde{\phi}_j^{(k)}, \tag{7}$$

which, substituting equations (5) and (6), has the closed form

$$\boxed{\Phi_j = \sum_{k=1}^{K} \underbrace{\frac{\mathrm{EAM}_k}{\sum_{m=1}^{K} \mathrm{EAM}_m}}_{\text{EAM weight}} \cdot \underbrace{\frac{1}{n}\sum_{i=1}^{n} \frac{|\phi_{i,j}^{(k)}|}{\sum_{m\in\mathcal{F}_k} |\phi_{i,m}^{(k)}|}}_{\text{mean L1-normalised magnitude}}} \tag{8}$$

The resulting vector Φ satisfies the following properties:

- $\Phi_j \in [0, 1]$ for all $j \in \mathcal{F}$.
- If all models share the same feature space, $\sum_j \Phi_j = 1$, i.e. the aggregation conserves explanatory mass.
- Φ is a magnitude-only representation; signed SHAP vectors should be inspected separately for direction-of-effect analysis.

5.6 QoS Algebra and Intent-Based Extension

Per-domain models produce KPI predictions independently for the RAN, Transport, and Core segments. To reason about end-to-end service quality, as required by an intent that specifies a global constraint such as maximum latency or minimum throughput, these domain-level predictions must be composed into a single E2E estimate. This composition is non-trivial: different KPIs obey different aggregation semantics (e.g. latency accumulates additively across hops, whereas throughput is constrained by the minimum-capacity domain). QoS algebra provides a principled, operator-driven framework for this composition making it a natural fit for the intent-assurance context.

To support intent-based network management, the per-domain KPI predictions are therefore combined using *QoS algebra operators* that estimate E2E KPI satisfaction. The remainder of this section is organised by KPI: throughput (Sec. 5.6.1), availability (Sec. 5.6.2), and latency (Sec. 5.6.3). Throughput is treated in most detail, as its non-smooth min-operator motivates three complementary explainers. A common KernelSHAP procedure for attributing composite predictions over the union feature space is described in Sec. 5.6.4.

5.6.1 Throughput

Throughput's QoS algebra operator is the minimum: end-to-end throughput cannot exceed the capacity of the most constrained domain. This operator is physically faithful but creates a discontinuity in the explanation space, the identity of the bottleneck domain can switch with arbitrarily small changes in inputs, causing abrupt changes in feature attributions even when the predicted value changes smoothly. Three complementary explainers are therefore provided: a minimum-routing explainer (Sec. 5.6.1), a black-box composite explainer (Sec. 5.6.1), and a soft-minimum approximation (Sec. 5.6.1).

Minimum Routing. The **minimum-routing** explainer is the most faithful option. For each instance, the domain with the smallest predicted throughput is identified as the bottleneck, and that domain's SHAP base value and feature contributions are presented as the E2E explanation directly:

$$T_{\text{e2e}} = \min_{d \in \mathcal{D}} M_{d,\text{tput}}(x_d), \tag{9}$$

where $d^* = \arg\min_d M_{d,\text{tput}}(x_d)$ is the bottleneck domain. This exactly reconstructs the minimum and produces a single-domain narrative of the form "the bottleneck is domain d^* because of features Y".

Black-Box Composite. The **black-box** explainer treats the composed min-operator as an opaque function. Analogously to the availability composite model, a single model maps the union of all domain features to the E2E throughput by taking the minimum of the three domain models' predictions. KernelSHAP is then applied to this composite function, estimating feature attributions by perturbing inputs and observing the resulting output changes:

$$T_{\text{e2e}} = f_{\text{bb}}(x_{\text{RAN}}, x_{\text{Trans}}, x_{\text{Core}}) \approx \min_{d \in \mathcal{D}} M_{d,\text{tput}}(x_d). \tag{10}$$

Because KernelSHAP perturbs the full input jointly, this explainer captures bottleneck-switching effects across features from all domains in a single attribution vector, at the cost of the higher computational overhead inherent to a model-agnostic method.

Soft-Minimum. The minimum-routing explanation can be unstable when two domains are nearly tied: a small perturbation may transfer the bottleneck label from one domain to another, producing a discontinuous jump in the attributed features even though the predicted throughput changes only marginally. The **soft-minimum** provides a differentiable approximation that keeps the explanation stable while still privileging the bottleneck domain:

$$T_{\text{soft}} = -\frac{1}{\tau} \log \sum_{d \in \mathcal{D}} \exp(-\tau \cdot M_{d,\text{tput}}(x_d)) \,, \tag{11}$$

where $\tau > 0$ is a temperature parameter; as $\tau \to \infty$, $T_{\text{soft}} \to T_{\text{e2e}}$. For explanation purposes, each domain's weight is set inversely proportional to its gap from the current minimum (with a small floor so the winner's weight remains finite), so the bottleneck receives the largest weight and near-tied domains share a smaller portion of responsibility. For example, given per-domain predictions of RAN = 21, Transport = 18, Core = 20 (Mbps), the gaps are $[3, 0, 2]$; after inversion and normalisation, Transport (the bottleneck) receives approximately 85–90 % of the weight, the near neighbour a smaller share, and the far domain almost none. The soft-blend explanation returns a weighted mixture of domain base values and feature contributions that reconstructs the soft prediction. The base value therefore differs between the minimum-routing and soft-minimum explainers by construction: the routing base is the bottleneck domain's expected value, whereas the soft-minimum base is a weighted average of all domain bases.

5.6.2 Availability

For availability, the E2E link is available only when all segments are simultaneously available. Assuming independence across domains, the composite operator is the product of per-domain availabilities:

$$A_{\text{e2e}} = \prod_{d \in \mathcal{D}} M_{d,\text{avail}}(x_d). \tag{12}$$

Unlike the throughput min-operator, the product is smooth and differentiable in each per-domain prediction, so a single composite explainer suffices.

5.6.3 Latency

For latency, the E2E delay is the sum of per-domain delays incurred as a packet traverses each segment:

$$L_{\text{e2e}} = \sum_{d \in \mathcal{D}} M_{d,\text{lat}}(x_d). \tag{13}$$

Additivity makes the latency operator the simplest of the three: domain contributions are linearly separable and the composite is smooth.

5.6.4 KernelSHAP for Composites

Since the composite functions (Eq. 9–13) are not tree-based, KernelSHAP is used to produce SHAP attributions over the *union feature space* $\mathcal{F}$. The background distribution against which the reference expectation $\mathbb{E}[g(X)]$ is estimated is a random sample of 100 instances drawn from the union test set, and each explained instance is attributed using 500 KernelSHAP coalition samples. This enables explanation of the composite prediction in terms of the original simulation parameters, providing operators with a single interpretable ranking over all features that explains the predicted E2E KPI.

6 System Design and Implementation

This section describes the prototype system developed to instantiate and evaluate the framework introduced in Section 5. The implementation is organised as a pipeline of loosely coupled stages, illustrated in Figure 2: simulation-based data collection (Section 6.1), per-domain model training (Section 6.2), a unified domain model interface (Section 6.3), cross-domain prediction and explanation orchestration (Section 6.4), QoS algebra composite predictors (Section 6.5), and the EAM-weighted aggregation procedure (Section 6.6). These last two stages address complementary axes of the problem: the QoS-algebra composites combine the per-domain predictions of a single KPI across domains using its physical composition operator, while the EAM-weighted aggregation combines the per-KPI attributions within a single domain into one domain-level profile. The design deliberately separates the framework's conceptual components from any specific model family or simulator, so that individual stages can be substituted without affecting the rest of the pipeline.

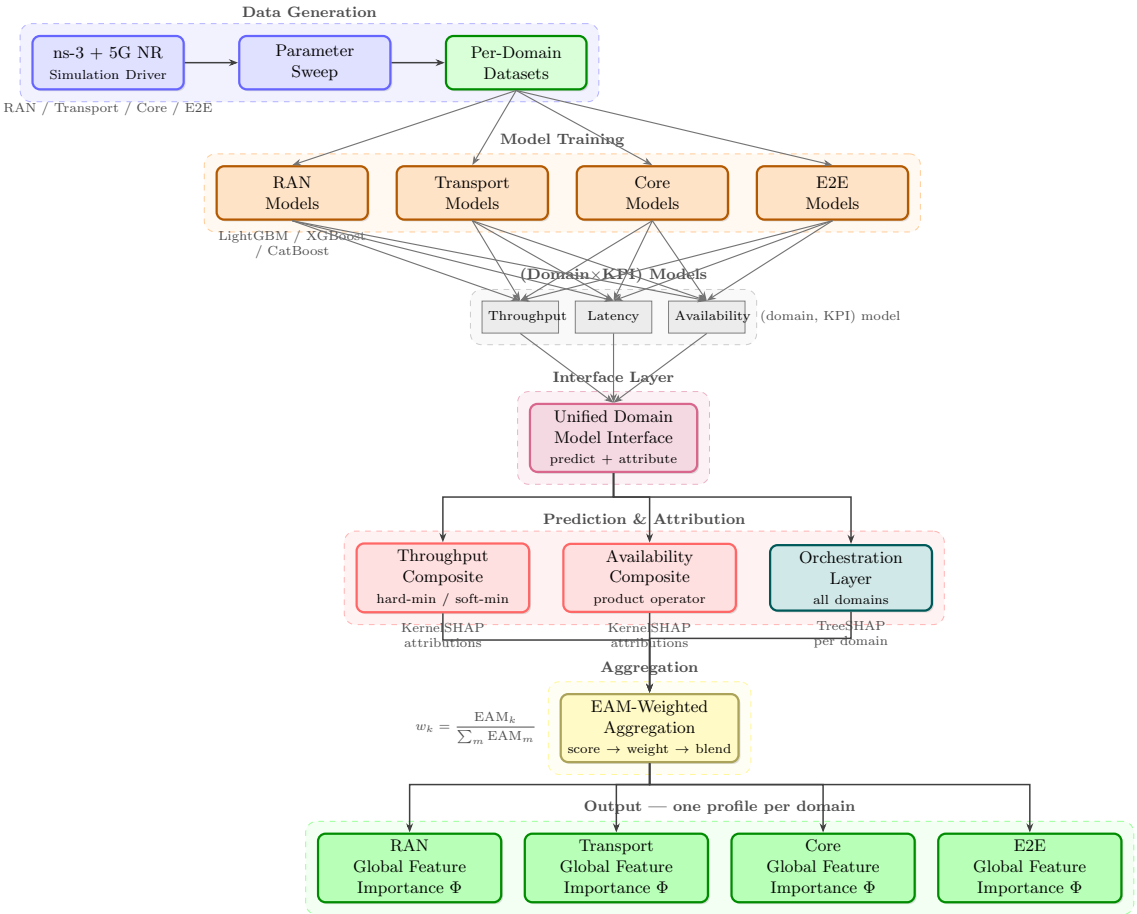


Figure 2: System pipeline. Simulation data are collected via ns-3, gradient-boosted tree models are trained for each (domain, KPI) pair, and attributions are computed via TreeSHAP (per-domain models) or KernelSHAP (composite predictors). The EAM-weighted aggregation stage produces one global feature importance profile Φ per domain (RAN, Transport, Core, and E2E).

6.1 Data Collection Pipeline

Training data is generated by the ns-3/NR simulation described in Section 5.1. A parameterised simulation driver exposes the configurable parameters of the RAN, Transport, and Core domains, and a sweep orchestrator runs the driver over a structured grid of parameter combinations.

The sweep configuration specifies the parameters that are varied and the range over which each is swept; the parameter sets used in this thesis are summarised in Table 3. A single joint sweep varies the RAN, Transport, and Core parameters simultaneously, and each simulation run emits four domain-scoped records: one for each of the three sub-domains (RAN, Transport, Core) and one for their end-to-end (E2E) composition. These records are then projected into four datasets. Because each per-domain record retains only that segment's swept parameters, single-domain models cannot observe cross-domain interactions by construction, whereas the E2E dataset spans the union of the sub-domain parameter spaces and so allows E2E models to capture those interactions.

Each simulation run thus contributes one data point per view, a vector of input parameters together with the resulting throughput, latency, and availability measurements, appended to the corresponding dataset. Datasets are then partitioned into training, validation, and test splits that are persisted to disk and reused across all downstream stages, ensuring that model selection and evaluation are performed against fixed, comparable splits.

6.2 Multi-Domain Model Training

For each (domain, KPI) pair, a regression model is trained to predict the KPI from that domain's parameters. To avoid committing the framework to a single learner, three representative gradient-boosted tree implementations, LightGBM, XGBoost, and CatBoost, are trained as candidates, and the candidate with the highest coefficient of determination (R^2) on the test set is selected as the model for that (domain, KPI) pair. All three implementations belong to the gradient-boosted tree family motivated in Section 2.2 and admit exact attribution via TreeSHAP, so model selection affects predictive quality without changing the explanation interface seen by downstream stages.

Training uses early stopping on the validation split to mitigate overfitting. The trained model is persisted together with a small metadata record describing the domain, target KPI, selected model family, and the feature columns on which the model was trained. This metadata is consumed by the model interface (Section 6.3) so that the rest of the system can remain agnostic to which model family was selected for any given (domain, KPI) pair. End-to-end models are trained on the E2E dataset using the same procedure, yielding one model per KPI defined over the union feature space.

6.3 Domain Model Interface

Each trained model is exposed through a unified interface that hides both the underlying model family and the model's specific feature subset. Given a (domain, KPI) pair, the interface loads the associated metadata, instantiates the correct backend, and exposes two capabilities:

- **Prediction.** Given an input instance defined over a (possibly larger) feature space, the interface projects the instance onto its model's feature subset and returns the predicted KPI value. This projection allows heterogeneous domain models, each defined over a different subset of the union feature space, to operate on a single shared input without conflict.
- **Attribution.** Given an input instance, the interface returns a structured attribution object containing the SHAP values, the model's base value ϕ_0, and the reconstructed prediction $\phi_0 + \sum_i \phi_i$. For the tree-based models used here, attributions are computed exactly via TreeSHAP.

Encapsulating predictions and attributions behind a common interface means that the orchestration and aggregation layers described next operate uniformly over all (domain, KPI) pairs and do not need to be modified when the underlying model family changes.

6.4 Cross-Domain Orchestration

The orchestration layer accepts the full collection of domain model interfaces, one per (domain, KPI) pair, and coordinates prediction and explanation across all of them. Given a network configuration expressed over the union feature space, it queries each model in turn and collects the per-(domain, KPI) predictions $\hat{y}^{(d,\kappa)}$ and attribution vectors $\phi^{(d,\kappa)}$, where d indexes the domain and κ indexes the KPI (throughput, latency, availability).

The orchestration layer does not itself perform any aggregation; it is purely a collection and dispatch component. The resulting two-dimensional grid of attribution vectors $\{\phi^{(d,\kappa)}\}_{d,\kappa}$ and the corresponding predictions feed the two downstream stages along the two different axes of this grid: for a fixed KPI κ, the per-domain predictions $\{\hat{y}^{(d,\kappa)}\}_d$ are combined across domains by the QoS-algebra composites of Section 6.5; for a fixed domain d, the per-KPI attributions $\{\phi^{(d,\kappa)}\}_\kappa$ are combined across KPIs by the EAM-weighted aggregation procedure of Section 6.6.

6.5 KPI Composition and Explanation Strategies

End-to-end KPIs are not produced by a single model; they arise from combining the per-domain predictions through KPI-specific algebraic operators. Because each operator has different mathematical properties, some additive, some multiplicative, some discontinuous, the corresponding explanation strategy must also differ. This subsection describes, for each of the three KPIs considered in this thesis, both how the end-to-end value is composed and how the corresponding explanation is produced.

Latency (sum). Latency composes additively across domains, so the end-to-end prediction is simply the sum of the per-domain latency predictions. Because Shapley attributions are themselves additive, the end-to-end explanation is obtained by summing the per-domain TreeSHAP attributions after aligning feature names over the union feature space. Both the base value and the per-feature contributions reconstruct the end-to-end latency exactly, so no black-box wrapping or model-agnostic explainer is required for this KPI.

Availability (product). Availability composes multiplicatively, under the assumption of independent failure modes across the RAN, Transport, and Core segments. Multiplicative interactions are not faithfully captured by summing per-domain additive attributions, since the contribution of a feature in one domain depends on the predicted availability of the other domains. The end-to-end availability function is therefore wrapped as a single black-box predictor over the union feature space: the wrapper invokes the three per-domain availability models and returns the product of their outputs. Attributions for this composite are computed using KernelSHAP, which perturbs the full input and measures the response of the product, thereby capturing shared features and cross-domain interactions natively. The output is in native availability units, and base-plus-contributions reconstructs the composed prediction by construction.

Throughput (min). Throughput composes through the minimum operator: the end-to-end throughput is bounded by the slowest domain. The min operator is discontinuous, and the identity of the bottleneck domain can switch under small perturbations of the input. Three explanation strategies are compatible with this algebra, of which the prototype implements the first:

- **Minimum routing (implemented).** For each instance, the domain with the smallest predicted throughput is identified and that domain's TreeSHAP explanation is surfaced directly as the end-to-end explanation. This reconstructs the min exactly and yields a crisp narrative of the form "the bottleneck is domain X because of features Y." It is the simplest and most faithful of the three options and is used throughout the evaluation section.
- **Black-box min composite (recognised).** The min function over the three domain predictors can be wrapped as a single black box over the union feature space and explained with KernelSHAP, analogously to the availability composite. This captures bottleneck-switching effects in the attribution surface itself instead of at the routing layer. It is not used in the prototype because minimum routing already provides exact reconstruction at lower computational cost, but it is retained as a deployable option for use cases in which a smooth attribution surface is preferred.
- **Soft-minimum blending (recognised).** Traditional minimum routing flips abruptly when two domains are nearly tied, which can destabilise the narrative under small input changes. A soft-minimum alternative weights each domain's explanation by its inverse gap to the current minimum (with a small floor so the bottleneck's weight remains finite) and blends the per-domain bases and contributions accordingly. The bottleneck retains the largest weight, but near-ties share responsibility. The framework's design admits this strategy, and EAM-derived trust weights could in principle be folded into the soft-minimum weights, but neither is implemented in the current prototype.

Local accuracy. For all three KPIs the chosen strategy satisfies local accuracy: the sum of base value and feature contributions reproduces the composed end-to-end prediction. This property is verified empirically in Section 7 as a sanity check on the composition pipeline.

6.6 EAM-Weighted Aggregation

The aggregation stage implements the EAM-weighted procedure formalised in Section 5.5. Conceptually, the orchestration layer produces a two-dimensional grid of attribution vectors $\{\phi^{(d,\kappa)}\}_{d,\kappa}$ indexed by domain d and KPI κ. EAM-weighted aggregation operates *within* a single domain, across its KPIs: fixing a domain d and combining the attributions $\{\phi^{(d,\kappa)}\}_{\kappa}$ across the three KPIs yields a single domain-level profile $\Phi^{(d)}$ that summarises which parameters drive the behaviour of domain d overall, irrespective of the specific KPI considered. This cross-KPI, intra-domain combination is the sole place in the system where EAM trust scores are used to weight a weighted sum.

The complementary cross-domain axis is instead composed by the QoS-algebra predictors of Section 6.5, without EAM weighting; the E2E model plays no part in either combination and serves only as an end-to-end benchmark.

Aggregation procedure. For a fixed domain d, the $K = 3$ per-KPI attribution vectors $\{\phi^{(d,\kappa)}\}_{\kappa\in\mathcal{K}}$ are combined; index these models $k = 1, \ldots, K$. The procedure consists of four steps:

1. **Quality scoring.** For each model k, the Fidelity F_k, Stability S'_k, and Consistency C_k scores are computed independently from the attribution vector and the corresponding model predictions, and combined into a single EAM score EAM_k.
2. **Weight derivation.** Aggregation weights are obtained by normalising the EAM scores, $w_k = \mathrm{EAM}_k / \sum_{m=1}^{K} \mathrm{EAM}_m$, so that more reliable models contribute proportionally more to the aggregated explanation.
3. **Alignment.** The three per-KPI models of a domain share that domain's feature set, so their attribution vectors are already defined over the same features. Each is L1-normalised and, for generality, expressed over the union feature space $\mathcal{F}$ (zero-padding any dimensions a model does not observe, which reduces to the identity in this intra-domain case), yielding aligned vectors $\tilde{\phi}^{(k)} \in \mathbb{R}^{|\mathcal{F}|}$.
4. **Aggregation.** The combined importance is computed as $\Phi_j = \sum_{k=1}^{K} w_k \, \tilde{\phi}_j^{(k)}$ for each feature $j \in \mathcal{F}$, and the resulting vector is ranked in descending order of magnitude.

The output is one domain-level profile $\Phi^{(d)}$ per domain, produced directly from the orchestration output without re-running any predictions or attributions. The full procedure is summarised in Algorithm 1, presented with its step-by-step description in Section 5.5.

Uniform-average baseline. A uniform-average baseline, in which all weights are set to $w_k = 1/K$, is computed alongside each EAM-weighted result. This baseline serves as the reference against which the contribution of EAM weighting is evaluated in Section 7.

7 Results

This section presents the experimental results in five layers, mirroring the structure of the framework: (1) per-(domain, KPI) model performance (Section 7.1); (2) the EAM component scores and the weights they induce (Section 7.2); (3) per-(domain, KPI) SHAP attributions and cross-model agreement (Section 7.3); (4) the two aggregation axes, cross-KPI within each domain via EAM weighting (Section 7.4) and cross-domain within each KPI via QoS-algebra composition (Section 7.5); and (5) the KPI-composition explanations produced by the strategies of Section 6.5 (Section 7.6). All results are computed on the held-out test split for each domain. The results reported here centre on the aggregation framework and its EAM. The ontology-based representation is illustrated conceptually in Section 7.8 using an illustrative vocabulary; its production schema (Section 1) is internal to the industrial project and is not empirically evaluated in this thesis.

7.1 Model Performance

Table 4 reports the test-set R^2 score for the best-selected model per (domain, KPI) pair, together with the model family selected for it (the candidate with the highest test-set R^2). Most cells exceed 0.93, confirming that gradient-boosted trees fit the structured tabular simulation data well.

Table 4: Test-set R^2 for the best model per (domain, KPI) pair. Best model family shown in parentheses. Domain ordering follows the data-flow convention (RAN → Transport → Core → E2E) used throughout this section.

KPI	RAN	Transport	Core	E2E
Throughput (Mbps)	0.9924 (CBst)	0.8332 (CBst)	0.9872 (CBst)	0.9691 (CBst)
Latency (ms)	0.9437 (LGBM)	0.9698 (CBst)	0.9867 (XGB)	0.6036 (CBst)
Availability (%)	0.9437 (CBst)	0.9663 (CBst)	0.9696 (CBst)	0.9271 (LGBM)

Two cells warrant comment. First, the RAN latency and RAN availability models share the same test R^2 to four decimal places (0.9437). This is a coincidental match of fit quality on this split rather than a shared mechanism: the two targets are driven by different attributions, latency chiefly by `ran_util_ratio` and availability by `snr_estimate_db`, as detailed in Section 7.3.

Second, the E2E latency model achieves a substantially weaker fit ($R^2 = 0.6036$) than every other cell. End-to-end latency is the sum of three domain-level latencies, each governed by qualitatively different physical regimes (propagation and scheduling in RAN, queueing in Transport, processing in Core), and the union feature space exposes the model to all three simultaneously. As a result, the prediction surface is less smooth than for any single domain, and, as shown in Section 7.2, this is reflected in the Stability component of the EAM score for that model.

7.2 Explanation Aggregation Metric Scores

Table 5 reports the Fidelity, Stability, and Consistency component scores, the combined EAM score, and the derived cross-KPI weight for each (domain, KPI) pair, computed *independently within each of the three aggregation domains* (RAN, Transport, Core). The End-to-End (E2E) model is not an aggregation domain: it is a

benchmark trained on the union feature space and is excluded from EAM weighting. Scores are computed on each domain's held-out test split by masking the top-3 SHAP features (replaced by their training-set column means) for Fidelity, adding Gaussian perturbations of magnitude $0.02\,\sigma_{\text{feat}}$ over 5 perturbation rounds for Stability, and averaging all pairwise cosine similarities across the test instances for Consistency, with a fixed random seed of 42. The best-selected model family per (domain, KPI) pair (Table 4) is used in each case.

Table 5: EAM component scores and derived *cross-KPI* weights, computed independently within each aggregation domain. Weights are normalised per domain so that $\sum_\kappa w_\kappa = 1$. E2E is a benchmark and is not aggregated.

Domain	**KPI**	**Fidelity** F_κ	**Stability** S'_κ	**Consistency** C_κ	**EAM** EAM_κ	**Weight** w_κ
RAN	Throughput	0.825	0.344	0.575	0.581	0.292
	Latency	0.334	0.872	0.672	0.626	0.315
	Availability	0.692	0.997	0.659	0.783	0.393
Transport	Throughput	0.934	0.927	0.517	0.792	0.348
	Latency	0.716	0.997	0.511	0.741	0.326
	Availability	0.711	0.997	0.516	0.741	0.326
Core	Throughput	0.768	0.449	0.589	0.602	0.293
	Latency	0.593	0.999	0.545	0.712	0.347
	Availability	0.708	0.997	0.511	0.738	0.360

Three observations stand out. First, the EAM scores are close within each domain, so the resulting cross-KPI weights are near-uniform: every weight lies within a few points of the uniform 1/3 baseline, and no KPI's attributions are strongly amplified or suppressed. This is the expected behaviour when all three per-KPI models fit their targets well (Table 4).

Second, the least-reliable KPI differs by domain and is driven by Stability: in RAN and Core it is *throughput* that is least stable ($S'_\kappa = 0.344$ and 0.449), whereas latency is highly stable within every aggregation domain ($S'_\kappa \geq 0.87$). The regime-switching that destabilises latency attributions appears only in the excluded E2E benchmark; Section 8.1 interprets these patterns.

Third, the Consistency component is comparatively uniform across KPIs and domains (0.51–0.67), so Stability is the primary differentiator on this dataset. Because the weights stay near-uniform, EAM weighting here acts as a mild reliability correction rather than a strong reweighting.

The weights in Table 5 are *cross-KPI* weights: within each aggregation domain they parameterise the combination of that domain's three per-KPI attribution vectors. As established in Section 5.5, this is the only place EAM weighting is applied; the cross-domain axis (Section 7.5) is composed by physical operators without reliability reweighting.

7.3 Per-(Domain, KPI) SHAP Attributions

This section summarises the raw per-(domain, KPI) attributions before any aggregation is applied. The full set of SHAP beeswarm plots for every (domain, KPI) pair, together with the cross-model-family comparison plots, is collected in Appendix B; the salient patterns are described here.

In the RAN domain (Figure 7), throughput is dominated by `offered_load_mbps` and `snr_estimate_db`, with `ran_util_ratio` also prominent, reflecting the roles of offered demand, radio-link quality, and radio-access utilisation in shaping radio-side throughput. RAN latency is driven chiefly by `ran_util_ratio`, followed by `packet_size_bytes` and `bw_per_ue_hz`, reflecting radio-access contention and per-user bandwidth. RAN availability is dominated instead by `snr_estimate_db`, with `ran_queue_factor` and `cell_radius_m` secondary.

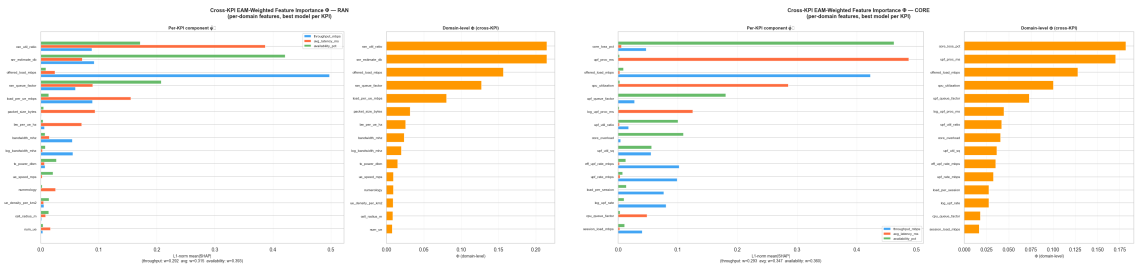
For the Core domain (Figure 9), `core_loss_pct`, `upf_proc_ms`, and `cpu_utilization` are the dominant attributions across the three KPIs, with `offered_load_mbps` also prominent, confirming that UPF processing delay, core packet loss, and compute utilisation are the principal drivers of core-side performance degradation. Transport-domain attributions (Figure 8) are dominated by `hop_total_delay_ms` and `backhaul_loss_pct`, with `offered_load_mbps` also prominent, consistent with the role of the backhaul as the queueing-bottleneck segment.

The E2E beeswarms (Figure 10) draw their top-ranked attributions from several segments at once: end-to-end throughput is led by backhaul and core offered-load features (`bh_offered_load_mbps` and `core_offered_load_mbps`), whereas end-to-end availability combines radio (`snr_estimate_db`), core (`core_loss_pct`), and backhaul (`backhaul_loss_pct`) drivers, confirming that end-to-end performance reflects the joint state of multiple network segments.

Finally, the cross-model-family comparison (Figure 11) shows that, within each domain, the SHAP feature-importance rankings produced by LightGBM, XGBoost, and CatBoost agree on the identity of the top-ranked features. This agreement between independently trained models supports the interpretation that those features capture genuine structure in the simulator, and justifies the use of the best-selected model's attributions as the per-(domain, KPI) explanation.

7.4 Cross-KPI Aggregation (per Domain)

The first aggregation axis fixes a domain d and combines the three KPI-specific attribution vectors $\{\phi^{(d,\kappa)}\}_\kappa$ into a single domain-level profile $\Phi^{(d)}$, as defined in Section 5.5. The weights used here are the cross-KPI EAM weights of Table 5.



(a) RAN (b) Core

Figure 3: Cross-KPI aggregated feature importance within the RAN and Core domains. Bar height reflects the EAM-weighted contribution of each feature across the three KPIs of that domain. The equivalent aggregation for the Transport domain follows the same qualitative pattern.

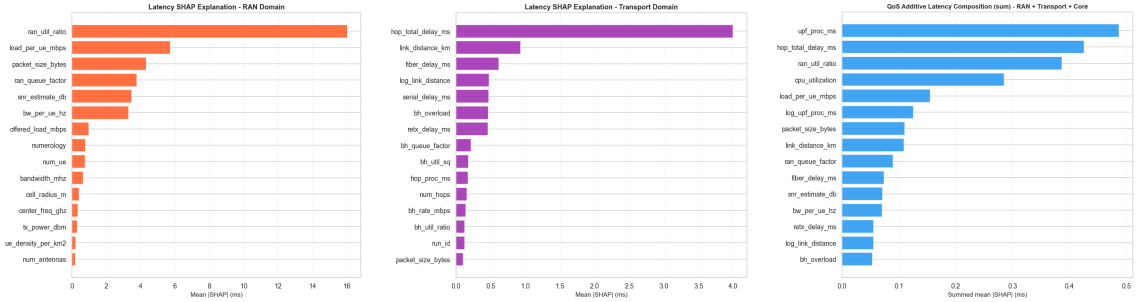
The cross-KPI profiles concentrate mass on the capacity-, load-, and quality-related features already prominent in the per-(domain, KPI) beeswarms: in the RAN domain, `ran_util_ratio`, `snr_estimate_db`, and `offered_load_mbps`

carry the most mass; in the Core domain, `core_loss_pct`, `upf_proc_ms`, and `offered_load_mbps` lead. Because the cross-KPI weights are near-uniform within each domain (Table 5), the aggregated profile closely tracks the unweighted mean of the three per-KPI attributions; the reliability correction reorders mid-ranked features slightly (for instance, the least-stable KPI in a domain has its distinctive attributions modestly attenuated) without displacing the dominant drivers.

7.5 Cross-Domain Aggregation (per KPI)

The second aggregation axis fixes a KPI κ and combines the three domain-specific predictions (RAN, Transport, Core) into a single end-to-end explanation for that KPI. As established in Section 5.5, this axis does *not* use EAM weighting: the three per-domain values of a single KPI are related by a known physical composition operator (Section 5.6), whereas no such law relates one KPI to another. These operators and their SHAP treatments are developed in Section 6.5; the resulting composed explanations are reported in Section 7.6.

The additive latency composition is the clearest illustration because it is exact: TreeSHAP attributions are additive, so the SHAP explanation of end-to-end latency is simply the sum of the per-domain latency attributions, with no approximation. Figure 4 shows the per-domain latency attributions (RAN and Transport) and the additive end-to-end composition side by side.



(a) RAN latency (b) Transport latency (c) Additive end-to-end latency

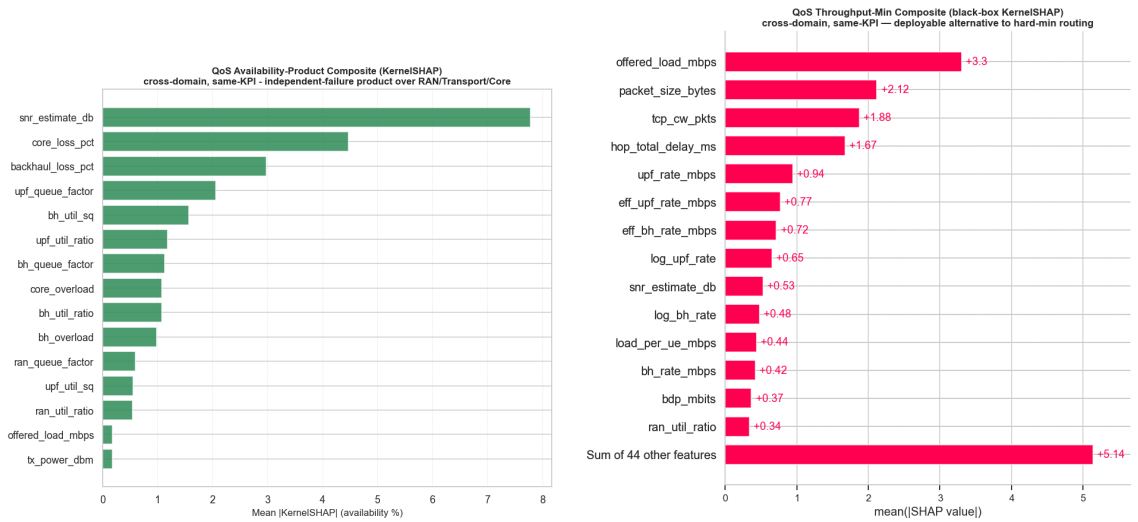
Figure 4: Cross-domain composition of latency by the additive operator. The per-domain RAN and Transport attributions (a,b) sum exactly into the end-to-end latency explanation (c); the Core contribution enters the same additive sum. Because TreeSHAP is additive, this composition introduces no approximation.

The composed latency explanation concentrates mass on the same drivers already prominent in the per-domain beeswarms of Section 7.3: the RAN contribution is dominated by `ran_util_ratio`, with `packet_size_bytes` and `bw_per_ue_hz` following, and the Transport contribution by `hop_total_delay_ms` and the associated delay-chain features. Summing these yields an end-to-end latency profile in which each domain contributes its own bottleneck features, so no single domain's drivers are amplified or suppressed by any weighting choice; the ranking reflects only the physical additive structure. The throughput (minimum) and availability (product) compositions are non-additive and are treated with the black-box strategies reported in Section 7.6.

7.6 KPI-Composition Explanations

This subsection reports explanations produced by the KPI-composition strategies of Section 6.5. Recall that each KPI has its own composition rule chosen to match its algebra: latency uses additive aggregation of per-domain TreeSHAP attributions; availability uses a black-box product wrapper explained with KernelSHAP; and throughput uses minimum routing in the prototype, with a black-box min wrapper retained as a deployable alternative. The two non-additive composites (throughput-min and availability-product) are explained with KernelSHAP over the union feature space; the throughput-min composite is reported as the deployable alternative discussed in Section 6.5, whereas the prototype's primary throughput explanation is produced by minimum routing, not by this composite.

Figure 5 shows the KernelSHAP attributions for the two non-additive composites over the union feature space: the availability-product composite (left) and the throughput-min composite (right). The KernelSHAP composite attributions are consistent with the per-domain TreeSHAP results of Section 7.3: load-related features dominate, and the composite captures cross-domain interactions such as the joint effect of high backhaul load and high UPF load on E2E throughput. This consistency supports the hypothesis that the QoS-composition operators correctly propagate the physical relationships captured by the per-domain models. It does not, on its own, validate the operators in any stronger sense: quantitative validation against simulator ground truth is left to future work.



(a) Availability-product composite

(b) Throughput-min composite

Figure 5: Cross-domain, same-KPI QoS-algebra composites explained with KernelSHAP over the union feature space. Left: end-to-end availability composed as the independent-failure product of the RAN, Transport, and Core availabilities. Right: end-to-end throughput composed as the minimum over domains (black-box min wrapper), reported here as the deployable alternative to minimum routing. No explanation-reliability weighting enters either composite; the composition is governed purely by the physical operator.

For latency, the additive KPI-composition rule means that the cross-domain composition of Figure 4 is itself the composed explanation, obtained by summing the per-domain latency attributions exactly. For throughput in the prototype, the minimum-routed explanation at any given instance is the TreeSHAP explanation

of whichever per-domain model has the smallest predicted throughput at that instance: Figure 6 shows one such instance, where the surfaced attribution is exactly the bottleneck domain's TreeSHAP explanation. Aggregate behaviour over the test split is therefore captured by the per-domain throughput beeswarms in Section 7.3.

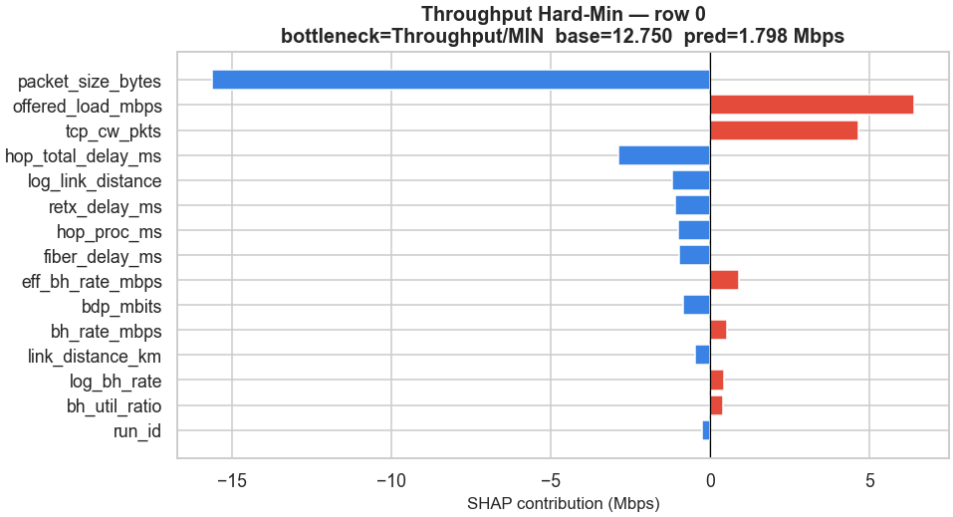


Figure 6: Throughput minimum routing at a single instance: the end-to-end throughput explanation is the TreeSHAP attribution of the bottleneck domain (the domain with the smallest predicted throughput), zero-padded to the union feature space. This is the prototype's primary throughput-composition strategy and reconstructs the minimum exactly.

7.7 Local Accuracy Check

As a sanity check on the composition pipeline, local accuracy was verified for each KPI on the test split: the sum of base value and feature contributions reproduces the composed end-to-end prediction to within numerical tolerance. For latency this holds by construction (additive aggregation of additive attributions). For availability, the KernelSHAP base-plus-contributions reconstructs the product to within the sampling error of the explainer. For throughput under minimum routing, the surfaced explanation is exactly the bottleneck domain's TreeSHAP explanation and reconstructs the domain prediction (and hence the min) exactly. No instances exceeded the local-accuracy tolerance on the test split.

7.8 Ontology-Based Representation of the Aggregated Explanations

For each domain, the framework produces a ranked vector of feature contributions and the per-KPI EAM scores that weight them, which in raw form are numeric arrays. To make them consumable by the intent-management layer, the aggregated explanations are serialised as an RDF graph, so that each contribution, reliability score, and domain profile becomes a typed, linkable resource. For example, the RAN cross-KPI result is encoded as a domain profile whose top feature contributions (`ran_util_ratio` and `snr_estimate_db`, $\Phi \approx 0.215$ each) and per-KPI EAM weights (throughput 0.292, latency 0.315, availability 0.393) become queryable properties keyed to shared feature and KPI resources.

This representation adds no explanatory content beyond the SHAP and EAM results above; it makes them *queryable* (which features drive a KPI, or which explanations exceed a reliability threshold), *linkable* (the per-domain and cross-domain profiles of Sections 7.4 and 7.5 share the same feature and KPI nodes, so an end-to-end explanation traces back to its domain-level contributions), and reliability-aware (the EAM score is a property of each explanation, so downstream logic can weigh how far to trust a contribution, not only its magnitude).

8 Discussion

This section interprets the results of Section 7 in light of the framework introduced in Section 5. Section 8.1 interprets the per-model Explanation Aggregation Metric (EAM) scores and the KPI-specific behaviour they reveal. Section 8.2 compares EAM-weighted aggregation against the uniform-average baseline. Section 8.3 examines two complementary forms of triangulation evidence: cross-model-family agreement within a domain, and cross-domain agreement on which features matter for each KPI. Section 8.4 discusses the explanations produced for the QoS-composition strategies of Section 6.5. Section 8.5 relates the findings to the related work reviewed in Section 3, and Section 8.6 draws the interpretation together into direct answers to the research questions. Sections 8.7–8.9 then step back from the specific results to reflect on the validation design, the strengths and trade-offs of the approach, and its transferability to other application domains. Section 8.10 closes with limitations and threats to validity, and links forward to the Conclusions section.

8.1 Interpretation of EAM Scores

The per-(domain, KPI) EAM scores reported in Table 5 reveal that reliability varies by KPI *and* by domain, which warrants careful interpretation. The scores should not be read as a direct quality ranking of the underlying predictive models: the test-set R^2 values in Table 4 already establish that the per-domain models fit the data competitively. EAM scores instead characterise the *reliability of the attributions* produced for each model, which is a separate axis.

Throughput and near-saturation instability. Within the RAN and Core domains, *throughput* receives the lowest Stability score ($S'_\kappa = 0.344$ and 0.449 respectively). This is not a calibration failure but a genuine structural property: throughput sits close to a capacity bottleneck, so in the near-saturation regime its prediction surface becomes steep and its SHAP attributions shift under small perturbations of the load-related features. The Stability component of the EAM correctly captures this. Because the effect is mild and the other two KPIs are comparably reliable, the resulting cross-KPI weights remain close to uniform.

Latency is stable per domain, unstable only end-to-end. Crucially, latency is *highly* stable within every aggregation domain ($S'_\kappa \geq 0.87$): each single-domain latency model operates on a smooth, regime-consistent surface (RAN scheduling, Transport queueing, or Core processing in isolation). The bimodal regime-switching that destabilises latency attributions arises only in the E2E benchmark, where the union model must span all three regimes simultaneously; this is consistent with the substantially weaker $R^2 = 0.6036$ of the E2E latency model (Section 7.1). Since the E2E model is excluded from aggregation, latency is *not* systematically down-weighted along the cross-KPI axis in any real domain. An operator who nevertheless wishes to prioritise a KPI can assign an explicit base weight α_κ, generalising the reliability-only weighting of Section 5.5 (which corresponds to the uniform case $\alpha_\kappa \equiv 1$ assumed there) to $w_\kappa = \mathrm{EAM}_\kappa\, \alpha_\kappa / \sum_{\kappa'} \mathrm{EAM}_{\kappa'}\, \alpha_{\kappa'}$: EAM-derived reliability and operator-defined priority are then orthogonal axes.

Availability and stability saturation. Availability attains near-maximal Stability across all three domains ($S'_\kappa \approx 1.000$). For core and transport this is expected: their availability predictions occupy an extremely narrow range (Section 2.3), staying close to 100 % with low variance under most operating conditions, so even substantial changes in load produce small prediction differences and correspondingly small SHAP perturbations. RAN availability, by contrast, is high-variance and driven by stochastic channel effects (Section 2.3), yet it too records $S'_\kappa \approx 1.000$. That even a volatile target saturates the score sharpens the methodological question this raises: whether Stability in its current formulation is a meaningful discriminator, given that it measures the change in attributions under small input perturbations without normalising by the prediction range. A variant that normalises attribution variance by the prediction range would be more informative in this regime; this is identified as a direction for future work in Section 8.10.

Consistency across KPIs. Consistency scores are broadly similar across KPIs and domains (0.51–0.67), indicating that each model exhibits comparable levels of cross-instance coherence in its feature rankings. This is consistent with the use of gradient-boosted tree ensembles (Section 2.2), which tend to learn relatively smooth prediction functions with stable feature rankings across the input space. The practical consequence is that, on this dataset, Stability is the primary differentiator between models, with Fidelity playing a secondary role and Consistency contributing a near-uniform offset. Because all three components sit in a narrow band within each domain, EAM weighting acts as a mild reliability correction, not a strong reweighting.

8.2 Weighted vs. Uniform Aggregation

Comparing the uniform-average baseline against the EAM-weighted cross-KPI aggregation within each domain (Figure 3 and Table 5) yields two main observations:

- The highest-ranked features are largely preserved between the two versions, confirming that the dominant aggregate signals are robustly identified regardless of the weighting scheme. This is the expected behaviour for features that genuinely drive multiple KPIs across multiple domains: their importance does not depend on the reliability profile of any single model.
- Mid-ranked features shift modestly under EAM weighting, consistent with the mild, domain-dependent reliability correction the cross-KPI weights encode. Because the per-domain weights are near-uniform (Table 5, all within a few points of the uniform 1/3), no KPI's attributions are strongly amplified or suppressed; the least-reliable KPI in a given domain (throughput in RAN and Core, driven by low Stability) has its distinctive signals proportionally attenuated, not eliminated.

Recall that EAM weighting is applied only along the cross-KPI axis *within* a domain; the cross-domain aggregations (Section 7.5) are composed by the physical QoS algebra of Section 5.6 and carry no reliability weights.

The practical implication is that EAM weighting provides a principled mechanism for attenuating the contribution of noisy or regime-switching explanations without

eliminating them entirely. This is the intended behaviour: the goal of the framework is to produce a unified explanation that reflects the relative reliability of its constituent attributions, not to discard low-reliability attributions outright.

8.3 Triangulation: Cross-Model and Cross-Domain Agreement

The results provide two complementary forms of triangulation evidence that strengthen the interpretation of the aggregated rankings.

Cross-model-family agreement within a domain. Figure 11 shows that, within each domain, the top-ranked features agree across LightGBM, XGBoost, and CatBoost. Independently trained model families converging on the same dominant attributions is evidence that those features capture genuine structure in the simulator and not artefacts of any one learning algorithm. The opposite outcome, substantial disagreement between model families, would be a warning sign of overfitting or of algorithm-specific feature interactions that do not generalise.

Cross-domain agreement on per-KPI drivers. The cross-domain per-KPI compositions of Section 7.5, formed by the QoS algebra rather than by any reliability weighting, show that `offered_load_mbps` is the top-ranked attribution for throughput in *all three* domains, a clear case of independent models converging on the same driver. For latency and availability the convergence is thematic rather than tied to a single shared feature: each domain contributes its own segment-specific driver of the same type, per-segment utilisation and delay for latency (`ran_util_ratio`, `hop_total_delay_ms`, `cpu_utilization`) and per-segment loss and utilisation for availability (`snr_estimate_db`, `backhaul_loss_pct`, `core_loss_pct`). Independent models trained on non-overlapping feature subsets converging on the same family of drivers is a stronger form of evidence than within-domain cross-model-family agreement, because the models do not even share an input space. This convergence aligns with the per-segment KPI definitions in Section 2.3: throughput is bounded by the minimum of per-segment effective rates, latency grows sharply as any segment approaches saturation, and availability degrades under sustained overload. All three KPIs are therefore expected to share a common dependence on the load–capacity balance at each segment, and the cross-domain rankings recover this expectation.

8.4 QoS-Composition Explanations

The KernelSHAP attributions for the black-box composite wrappers (Figure 5) are consistent with the per-domain TreeSHAP results but provide an additional cross-domain perspective. For the throughput-min black-box wrapper, attribution mass concentrates on features belonging to the *bottleneck* domain: when one segment is closer to saturation than the others, its features dominate the composite's attributions. This bottleneck-attribution property is exact for the prototype's minimum-routing strategy (which surfaces the bottleneck domain's TreeSHAP explanation directly) and approximate for the KernelSHAP wrapper (which estimates it via sampling over the union feature space). Both strategies are directly useful for intent-violation diagnosis: a composite explanation concentrating attribution on

transport-domain features immediately identifies the backhaul as the limiting segment, without requiring the operator to inspect each per-domain model individually.

Two caveats apply, separated by concern.

Computational cost. KernelSHAP on composite wrappers is substantially more expensive than TreeSHAP because it requires many evaluations of the composite function over a sampled feature-coalition space. For real-time deployment in a production network management system, more efficient approximations or pre-computed explanation caches would be needed. The minimum-routing strategy avoids this cost in the throughput case by reusing per-domain TreeSHAP attributions directly.

EAM comparability. Composite attributions produced by KernelSHAP are sampling-based and not exact. As a consequence, EAM scores computed for composite explanations are not directly comparable to those computed for the underlying TreeSHAP attributions of the per-domain models; any extension of the EAM framework to composites would need to account for this difference explicitly, for example by reporting sampling-error confidence intervals on the Stability and Fidelity components.

8.5 Relation to Prior Work

Relative to the gap identified in Section 3 (Table 2), the contribution of this thesis is to combine three elements not previously brought together in a single framework: (i) aggregation across *physically distinct domains with non-overlapping feature subsets*, in a centrally trained (non-federated) setting where each domain model is trained on simulator ground truth; (ii) an explicit *reliability-weighted* combination rule that shares the goal of Bello et al. [12], reducing explanation quality to a computable score, but replaces their single scalar for single-model classification with a multi-axis (fidelity, stability, consistency) reliability measure for the multi-model regression setting; and (iii) an *intent-oriented* presentation layer that ties the aggregated explanation back to operator-facing QoS KPIs via the QoS algebra of Section 5.6.

EAM-weighted and uniform aggregations here agree on their highest-ranked features while differing in the middle of the ranking: dominant signals are recovered by either combination rule, whereas the ordering of moderately important features is more sensitive to the weighting scheme, which is where reliability-aware weighting has its main effect. This is consistent with the premise of Vascotto et al. [37], who aggregate explanations from multiple independently trained models and treat inter-model agreement as a trustworthiness signal. Their setting assumes interchangeable models over a shared input space; the pattern is observed here in a multi-domain setting where the constituent models are instead defined over non-overlapping feature subsets.

8.6 Answering the Research Questions

The interpretation above can now be drawn together into direct answers to the three research questions posed in Section 1.

RQ1: How can SHAP attributions from heterogeneous 5G domain models be aggregated? The framework answers this with a two-axis aggregation scheme (Section 5.5). The *intra-domain, cross-KPI* axis combines a domain's three per-KPI attribution vectors, which already share that domain's feature set, by an EAM-weighted convex combination of their L1-normalised shares, whereas the *cross-domain, same-KPI* axis is composed into operator-facing end-to-end KPIs by the QoS algebra of Section 5.6 rather than by reliability weighting; alignment onto the union feature space is required only on this cross-domain axis, where the domains genuinely differ in their features. The separation is deliberate: KPIs within a domain obey no physical law relating one to another and are therefore combined by explanation quality, whereas the per-domain values of a single KPI are related by a known physical operator (sum, minimum, product) and must be composed by that operator. The cross-domain agreement examined in Section 8.3, where independently trained models on non-overlapping feature subsets converge on the same load-related drivers, is evidence that this scheme aggregates genuine network structure rather than algorithm-specific artefacts.

RQ2: How can explanation reliability be quantified? Reliability is quantified by the Explanation Aggregation Metric, the unweighted mean of Fidelity, Stability, and Consistency, each bounded in $[0, 1]$ (Section 5.4). As analysed in Section 8.1, the empirical scores are close within each aggregation domain, so the induced cross-KPI weights are near-uniform; Stability is the primary differentiator on this dataset, with Fidelity secondary and Consistency contributing a near-uniform offset. The least-reliable KPI differs by domain and is driven by Stability, namely throughput in RAN and Core, which sit near a capacity bottleneck. Latency, by contrast, is stable within every single domain and destabilises only in the excluded E2E benchmark, where one union model must span three regimes at once. Availability attains near-maximal Stability, which Section 8.1 shows partly reflects its narrow dynamic range instead of superior attribution quality, motivating the range-normalised Stability variant identified in Section 8.10.

RQ3: How does EAM-weighted aggregation differ from uniform averaging, and is the difference principled? As shown in Section 8.2, the two aggregations agree on their highest-ranked features, a triangulation check that the dominant signals are robust to the weighting scheme, and differ in the middle of the ranking, where EAM weighting proportionately attenuates the least reliable per-KPI contributors without discarding them. Because the per-domain weights stay within a few points of the uniform 1/3 (Table 5), the correction is mild on this dataset. The difference is therefore principled rather than performance-optimising: it attenuates noisy contributors in proportion to a measurable property of their attributions. This also delimits the present evidence. Because the models here all fit their targets well, the regime in which reliability weighting would matter most, a genuinely unreliable model sitting among reliable ones, is not strongly exercised by this dataset; that regime, and independent confirmation of the metric itself, are precisely where further validation is needed (Section 8.10).

8.7 Validation Design and Evidentiary Basis

A methodological point underlies every claim above and deserves to be made explicit: there is no ground-truth feature attribution to validate against. SHAP explains the behaviour of a fitted model, and the ns-3 simulator exposes KPI outcomes, not the "true" importance of each input to those outcomes. No oracle labels a given attribution as correct, so the framework cannot be evaluated by direct comparison against known-correct explanations. The validation strategy adopted here is therefore one of *convergent evidence*: several independent checks, each fallible on its own, are made to bear on the same conclusion so that agreement between them is difficult to explain except by the aggregated rankings capturing genuine structure.

Four checks make up this design, in increasing evidentiary strength. Predictive fidelity comes first: the per-domain models attain high test-set R^2 (Table 4), establishing that they have learned the input–output relation well enough to be worth explaining at all. Cross-model-family agreement (Figure 11) then tests whether the surfaced attributions reflect the simulator or the learner: LightGBM, XGBoost, and CatBoost converge on the same top features within each domain, which would be unlikely if those features were algorithm-specific artefacts. Cross-domain agreement on per-KPI drivers (Section 8.3) is stronger still, because models trained on *disjoint* feature subsets recover the same load–capacity story that the per-segment KPI definitions of Section 2.3 predict on physical grounds. Finally, the local-accuracy checks of Section 7.7 confirm that the composition pipeline reconstructs the composed predictions to numerical tolerance, so the cross-domain explanations are arithmetically faithful to the quantities they claim to explain.

This design establishes what convergent evidence can establish: that the aggregated rankings track reproducible, physically interpretable structure, and that the composition machinery is internally sound. It cannot, by construction, establish two things. It cannot confirm that Fidelity, Stability, and Consistency are the correct and complete set of reliability dimensions, since that is a claim about the EAM *construct* for which no simulator-derived referent exists (Section 8.10). And it cannot show that a high-EAM explanation is *useful* to a human operator, which is a question about the presentation layer answerable only by user study. The evaluation is also deliberately conservative in one further respect noted in Section 8.6: because the constituent models all fit their targets well, the reliability-weighting mechanism is exercised in a favourable regime and not a stress regime in which a genuinely unreliable model competes with reliable ones. The validation therefore speaks to the soundness of the mechanism, not yet to its discriminating power under adversarial conditions.

8.8 Strengths and Trade-offs of the Approach

Stepping back from the individual results, the framework has a characteristic profile of strengths and compensating costs that is worth stating plainly, since several of the design choices buy one property at the expense of another.

Strengths. The two-axis design (Section 5.5) matches the structure of the problem: it applies reliability weighting only where no physical law relates the quantities being combined (the cross-KPI axis) and defers to a known algebra wherever one exists (the cross-domain axis). A direct consequence is that physically meaningful

compositions stay exact, and sampling approximation is confined to the two non-additive composites where it is unavoidable. The use of exact TreeSHAP on the tree models reinforces this: the reliability signal the EAM measures is not confounded by explainer noise (Section 2.2), a property that would be lost with a sampling-based attributor. The framework is also model-agnostic (EAM scoring and aggregation are decoupled from the choice of learner, so any per-domain model that admits attributions can be substituted), and its magnitude-preserving, L1-normalised aggregation yields a numerical signal that can be thresholded for intent-assurance decisions, which rank-based aggregation could not supply (Section 3). Finally, low-reliability contributions are proportionally attenuated instead of discarded, so no signal is lost outright.

Trade-offs and costs. Several of these strengths have a price. The equal weighting of Fidelity, Stability, and Consistency and the use of *static* EAM weights are simplifying choices that favour interpretability and reproducibility over adaptivity; a tunable or dynamically re-estimated weighting could track distribution shift but would be harder to reason about and to reproduce (Section 8.10). The value of reliability weighting is inherently regime-dependent: when the constituent models are uniformly reliable, as on this dataset, the weighted aggregation approaches the uniform average and the mechanism earns little, so its benefit is realised only when model reliability genuinely varies. The KernelSHAP composites are computationally expensive and their EAM scores are not directly comparable to the exact TreeSHAP scores (Section 8.4). And the cross-domain axis depends on knowing the composition operators: the QoS algebra encodes domain knowledge that must be supplied and is not learned, so the approach trades generality for the correctness that this knowledge guarantees where it is available.

8.9 Transferability to Other Application Domains

Although this thesis is developed and evaluated entirely in a 5G network setting, the organising principle of the framework is not specific to telecommunications. The essential structure is a system that decomposes into subsystems, each described by its own predictive model over a distinct feature subset, whose outputs must be combined into a system-level explanation along two qualitatively different axes: one where the quantities obey no known composition law and are best reconciled by explanation quality, and one where a known operator relates them and should govern the composition exactly. Any domain exhibiting this structure is a candidate for the same treatment.

Several settings fit the pattern. End-to-end latency in a multi-tier microservice architecture composes additively along a request's critical path, much as segment latencies sum here; service availability in a supply chain or a distributed system composes as a product of per-stage reliabilities, mirroring the availability operator; and a bottleneck resource in a pipeline behaves like the throughput minimum. Multi-sensor fusion and multi-modal clinical models, where separate models observe non-overlapping feature sets and a clinician needs one consolidated, reliability-aware explanation, share the cross-model aggregation problem even where no closed-form composition operator exists. In that last case the cross-subsystem axis would fall back to reliability weighting, since the algebra that governs the cross-domain axis

here would be absent.

Three preconditions delimit where the approach transfers cleanly: a meaningful decomposition into subsystem models must exist; the cross-subsystem axis needs either a known composition operator or an accepted reason to treat the subsystems as reliability-weighted views of a shared quantity; and the attributions must be placed on a comparable scale, which is what the L1-normalisation step provides. What does not transfer unchanged is the domain knowledge itself: the specific QoS operators and the particular choice of reliability dimensions are tailored to 5G KPIs, so applying the framework elsewhere means re-deriving the composition algebra for the new domain and revisiting whether Fidelity, Stability, and Consistency remain the reliability dimensions that matter.

8.10 Limitations and Threats to Validity

The evaluation in this thesis is subject to several limitations that also define directions for future work presented in Section 9.2. We group them by the standard internal/external/construct validity framing.

External validity.

1. **Simulation data only.** All experiments use ns-3 with the NR module (Section 2.3). While the simulator provides a controlled and physically motivated environment, it does not capture every aspect of real deployments. Examples include fine-grained interference patterns, handover dynamics, and hardware variability. Validation on real network measurement data is a necessary next step.

2. **Single topology.** The data generation sweeps described in Section 6.1 use a single base topology. Multi-cell scenarios with inter-cell interference and mobility-induced handovers would require a more complex simulation configuration and potentially different feature engineering.

Internal validity.

3. **Static EAM evaluation.** EAM scores are computed once on the held-out test set and used as fixed weights. In a deployed network, distribution shifts (traffic growth, topology changes, seasonal effects) will alter explanation reliability over time. A dynamic re-evaluation mechanism that periodically recomputes EAM scores and updates aggregation weights is needed for production deployment.

4. **Top-k masking for Fidelity.** The Fidelity computation masks the top-3 features by absolute attribution value. The choice $k = 3$ is a hyperparameter; different values may yield different fidelity estimates, and the sensitivity of the EAM score to this choice has not been systematically evaluated.

5. **KernelSHAP approximation.** Composite-model attributions are computed via KernelSHAP, which is a Monte Carlo approximation. The resulting EAM scores for composites are therefore not directly comparable to the exact TreeSHAP-based EAM scores of the domain models, as discussed in Section 8.4.

Construct validity.

6. **Stability normalisation.** The current Stability formulation does not normalise attribution variance by prediction range. For KPIs with very narrow dynamic range (such as availability), this makes Stability a less discriminative signal and inflates the score in a way that does not reflect a meaningful difference in attribution reliability. A range-normalised variant would be more informative in such cases and is the most immediately actionable methodological improvement identified by this work.

7. **Reliability versus usefulness.** EAM measures technical properties of attributions (fidelity, stability, consistency). It does not measure whether the resulting explanations are *useful* or *understandable* to a human operator. A high-EAM explanation pointing at `bh_util_ratio` is no more comprehensible than a low-EAM one if the operator does not know what backhaul load ratio means in their operational context. Closing this gap requires user-study evaluation, which is outside the scope of this thesis.

8. **Unvalidated metric construct.** The EAM is a *constructed* measure. The choice of Fidelity, Stability, and Consistency as its three dimensions, and their combination as an equally weighted mean, is a principled design grounded in the explanation-quality literature (Section 3), but it has not been empirically confirmed as a definitive measure of attribution reliability. Whether these three dimensions are the right ones, whether they should be weighted equally, and whether a high EAM score corresponds to what practitioners regard as a trustworthy explanation are open questions that the present evaluation, conducted entirely against simulator ground truth, cannot settle. Establishing the construct validity of the metric requires evaluation and confirmation with network operators and domain experts; this is set out as future work in Section 9.2. Until then, EAM should be read as a well-motivated reliability signal rather than a proven, definitive measure of explanation quality.

9. **Undisclosed companion component.** The ontology-based explanation representation that was also designed and implemented within the wider industrial project (Section 1) is proprietary and cannot be disclosed. Its production schema and vocabulary are therefore withheld, and the claims of this thesis are confined to the reliability-aware aggregation framework and its EAM, which are reported and evaluated in full. The unavailability of that artefact limits external reproducibility of the end-to-end intent-to-explanation pipeline, though not of the aggregation results presented here.

These limitations are revisited in Section 9, which summarises the contributions of the thesis and outlines directions for further work. Ethical and governance considerations for deploying the framework are discussed separately in Section 10.

9 Conclusions and Future Work

This thesis addressed the problem of aggregating SHAP feature attribution explanations across multiple heterogeneous predictive models in a multi-domain 5G network. The core challenge is that explanations from different models, trained on different domains, predicting different KPIs, and operating on different feature subsets, are not directly comparable or combinable without a principled framework. The contribution of this work is a reliability-weighted aggregation pipeline that produces a single, operator-facing feature-importance profile from a heterogeneous collection of per-domain SHAP attributions, together with an intent-aligned composition layer that ties the aggregated explanation back to end-to-end QoS KPIs.

The three research questions posed in Section 1 are taken up and answered in full in Section 8.6, where the two-axis aggregation scheme (RQ1), the EAM as a bounded measure of explanation reliability (RQ2), and the comparison of EAM-weighted against uniform aggregation (RQ3) are each discussed against the empirical results. This section does not repeat those answers; it summarises the contributions the thesis delivered and the directions they open.

The framework was implemented end-to-end, covering network simulation for data collection, multi-domain training of gradient-boosted tree models, TreeSHAP attribution, EAM evaluation, reliability-weighted intra-domain (cross-KPI) aggregation, and cross-domain composition via the QoS algebra. A prototype orchestration engine was developed that orchestrates domain-level predictions and composes them into KPI-level explanations via the QoS algebra, with the attribution strategy matched to each operator (exact TreeSHAP where the composition is additive or routed, KernelSHAP where the composite is treated as a black box).

9.1 Key Contributions

The main contributions of this thesis are:

1. **A reliability-aware SHAP aggregation framework** for multi-domain 5G networks that combines L1-normalised attribution vectors through two complementary mechanisms: an EAM-weighted convex combination on the intra-domain, cross-KPI axis, and QoS-algebra composition (additive, minimum, product) on the cross-domain axis. The framework satisfies the normalisation, reliability-weighting, and interpretability requirements identified in the problem formulation, and its cross-KPI aggregation admits a generalised form $w_\kappa = \mathrm{EAM}_\kappa\, \alpha_\kappa / \sum_{\kappa'} \mathrm{EAM}_{\kappa'}\, \alpha_{\kappa'}$ in which operator-defined intent priorities α_κ and EAM-derived reliability are expressed as orthogonal factors. Because the attribution vectors are L1-normalised before combination, the aggregated profile expresses *relative* explanatory importance (each feature's normalised share of the total attribution mass) and not absolute global importance.

2. **A formal Explanation Aggregation Metric (EAM) for SHAP explanations**, comprising three operationalised quality dimensions (Fidelity, Stability, Consistency) each bounded in $[0, 1]$, with formal definitions and an empirical evaluation on gradient-boosted tree models for 5G KPIs.

3. **A multi-domain 5G simulation dataset** across RAN, Transport, and Core domains generated via network simulation, with a rich feature set including

engineered load-ratio and interaction features. Per-domain row counts and the exact sweep configuration are reported in Section 6.1.

4. **A prototype orchestration engine** that orchestrates per-domain models and composes them via the QoS algebra (additive, minimum routing, product) with attribution strategy matched to operator type (TreeSHAP where exact, KernelSHAP where the composite is treated as black-box), providing a complete pathway from domain-level models to intent-aligned E2E explanations.

5. **Validation that the framework recovers physically interpretable cross-domain drivers**: `offered_load_mbps` is top-ranked for throughput across all three domains, and each domain contributes its own segment-specific utilisation, delay, and loss features for latency and availability, matching the per-segment KPI definitions of Section 2.3 and providing triangulation evidence that the aggregated explanations capture genuine structure in the simulator, not algorithm-specific artefacts.

9.2 Future Work

Several directions for future work are identified, grouped by whether they extend the methodology, the empirical scope, or the deployment path.

Methodological extensions. *Range-normalised Stability.* Normalising attribution variance by prediction range, addressing the limitation identified in Section 8.10, is the most immediately actionable methodological improvement identified by this work.

Top-k sensitivity for Fidelity. A systematic study across the masking depth k (Section 8.10), possibly with an adaptive choice based on attribution-mass concentration, would strengthen the construct validity of the Fidelity component.

Regime-conditional Consistency. The Consistency metric penalises models that legitimately produce different explanations in different operating regimes. Extending Consistency to a regime-conditional variant, clustering test instances by operating regime and computing cosine similarity within clusters, would provide a more informative quality signal for multi-regime KPIs such as latency.

Feature-level EAM. The current EAM is model-level: a single scalar summarises the explanation quality of an entire model. A feature-level extension, where each feature j receives its own fidelity, stability, and consistency score, would enable more targeted uncertainty quantification and could identify features whose attributions are reliable across models versus those that are noisy.

Composite-explanation EAM. Because KernelSHAP-based composite EAM scores are not comparable to the exact TreeSHAP-based ones (Section 8.10), a composite-aware variant reporting sampling-error confidence intervals on Stability and Fidelity is needed before composites can join the same reliability ranking as their constituent models.

Independent evaluation of the aggregated profile. The comparison in Section 8.2 shows that EAM-weighted and uniform aggregation agree on their highest-ranked features and differ mainly among the middle-ranked ones. Deciding which of the two better reflects true feature importance calls for evaluating the *final aggregate* against yardsticks independent of the EAM itself, such as ablation- or retraining-based

importance, sensitivity to input resampling, or agreement with simulator-derived perturbation studies. Such external metrics would turn the observed weighted-versus-uniform differences into a measurable quality judgement, going beyond the internal-consistency and triangulation evidence available here (Section 8.7).

Causal attribution. SHAP values are associative, not causal. Extending the framework with causal discovery methods (e.g., structural causal models fitted to network telemetry) would allow attributions to be interpreted as causal scores, enabling more actionable closed-loop recommendations.

Empirical-scope extensions. *Real network data validation.* Validating the framework on real operator telemetry, not the simulation data used here (Section 8.10), is essential for assessing whether the identified features generalise to production environments and whether EAM scores remain discriminative under real measurement noise.

Multi-cell and mobility scenarios. Moving beyond the single base topology of the current sweeps (Section 8.10), multi-cell scenarios with inter-cell interference and mobility-induced handovers would test whether the cross-domain feature rankings remain stable in more complex deployments.

Empirical evaluation of intent-driven α_κ weights. The generalised aggregation formula already admits operator-declared intent priorities α_κ alongside EAM-derived reliability. The mechanism is part of the framework, but its empirical impact under realistic intent profiles (URLLC-prioritised, eMBB-prioritised, mMTC-prioritised) was not evaluated and is a natural next step.

User evaluation and validation of the aggregation metric. EAM measures technical properties of attributions, and its three dimensions and equal weighting are a reasoned design, not an empirically confirmed optimum (Section 8.10). Two questions therefore remain open and motivate a dedicated study with network operators and domain experts. The first is *construct validity*: whether the EAM score agrees with expert judgements of which explanations are trustworthy, and whether its dimensions or their weighting should be revised or extended. The second is *usefulness*: whether high-EAM explanations are also more understandable and actionable in practice. Such an evaluation would move the metric from a principled construct towards a confirmed one and would inform the design of the presentation layer above the aggregation pipeline. Until it is carried out, EAM is best read as a well-motivated reliability signal, not a definitive measure of explanation quality.

Deployment-path extensions. *Dynamic EAM evaluation.* EAM scores are currently computed statically (Section 8.10), but distribution shifts in a deployed network will alter explanation reliability over time. A dynamic re-evaluation mechanism that periodically recomputes EAM scores and updates aggregation weights, coupled with concept-drift detection on the underlying models, is a research-and-engineering direction towards production deployment.

Online inference and explanation caching. The current prototype loads models from disk and computes SHAP values on demand. For real-time intent verification in an O-RAN xApp or rApp context, a streaming inference and explanation pipeline with latency guarantees would be required. KernelSHAP's computational cost (Section 8.4) is the binding constraint for the composite operators; pre-computation of

attributions for representative configurations together with real-time interpolation or surrogate models is a plausible engineering path.

9.3 Closing Remarks

The broader motivation for this work is the trajectory of mobile networks towards autonomous, intent-driven operation. As control loops increasingly close around machine-learned predictions, the quality of the explanations that accompany those predictions becomes operationally significant: an explanation that is unstable, unfaithful to the model, or inconsistent across instances will mislead the operators and automation layers that consume it. The framework presented here treats explanation reliability as a first-class quantity, measurable per model and propagated through the aggregation rule. The reliability score is not a certificate of correctness, and no single metric can replace operator judgement in safety-critical deployments; but making reliability explicit, weightable, and auditable is, we argue, a necessary precondition for the kind of trustworthy explanation pipeline that intent-based 5G management will ultimately require.

10 Research Ethics and Data Confidentiality

All experimental evaluations in this work are conducted using simulation data generated by the ns-3 network simulator with its 5G NR module (Section 2.3); no real, operational, or user-derived network data are used. Consequently, no personally identifiable information (PII) is collected, stored, or processed at any stage of the research.

No industrial or proprietary *datasets* are used; all data are simulated as described above.

Confidentiality of the ontology component. The ontology-based explanation representation (Section 1) was designed and implemented within the wider industrial project in which this thesis was conducted, but its schema, vocabulary, and internal evaluation are proprietary to that organisation and are subject to corporate confidentiality restrictions. For this reason the component is described only conceptually, and *no evaluation of it is reported in this thesis*: the absence of an ontology evaluation reflects this company restriction. The contributions that are reported and evaluated in full are the reliability-aware aggregation framework and its Explanation Aggregation Metric, which are entirely the author's own work and are not subject to these restrictions.

Source code, models, and experimental artefacts developed in this thesis are maintained in controlled repositories and shared only in accordance with institutional and corporate policies. Where open-source components are used, their licences and ethical usage guidelines are respected.

10.1 Ethical Implications of the Research Outcomes

The primary ethical objective of this work is to enhance transparency, accountability, and trustworthiness in autonomous network management systems. By enabling system-level explainability, the proposed framework supports responsible AI deployment and helps mitigate risks related to opaque automated decision-making. Improved explainability contributes to ethical system governance by facilitating auditing, compliance verification, and human oversight, reducing the likelihood of undetected harmful behaviour, unintended bias, or misaligned system actions, and reinforcing human responsibility in critical network operations. Several specific considerations nevertheless apply.

Explanation over-trust. Feature attribution explanations may be misinterpreted as causal statements. A high SHAP value for `bh_util_ratio` indicates that this feature is predictive of the KPI within the model's learned function; it does not, on its own, imply that reducing backhaul load will improve the KPI proportionally in the deployed network. Operators must be trained to interpret attributions as associative and not causal, and any closed-loop control action triggered by an explanation should be validated against an independent causal model or against operational experience.

Automation bias. The EAM mechanism provides a principled basis for differential weighting of explanations, but the resulting weights can be misinterpreted as a certificate of correctness. The risk is that high-EAM models, whose attributions

dominate the operator-facing aggregated explanation, are under-scrutinised on the assumption that the framework has already validated them; in fact, EAM measures only the reliability of the *attribution*, not the correctness of the underlying *prediction*. A model can be highly stable, internally consistent, and locally faithful while still being systematically biased on out-of-distribution inputs. In safety-critical deployments, human review of high-EAM explanations is therefore as important as review of low-EAM ones, and EAM should be presented to operators as a diagnostic aid rather than as a trust score.

Interpretability is not certified. Relatedly, the EAM does not certify human-interpretability. Two explanations with identical EAM scores can differ substantially in how intelligible they are to a non-expert operator. Any production deployment of the framework should pair the technical reliability score with a separate evaluation of presentation-layer comprehensibility, ideally informed by user studies with the intended operator population.

Data governance. As noted above, all data used in this thesis are from simulation, so no personally identifiable information is involved.

References

[1] "Autonomous networks with multi-layer, intent-based operation," Ericsson Technology Review, 2021, accessed: Feb. 2026. [Online]. Available: https://www.ericsson.com/en/reports-and-papers/ericsson-technology-review/articles/autonomous-networks-multi-layer-intent-based

[2] B. Bilgin, S. Aktaş, and S. Baş, "Intent-based end-to-end explainability orchestration framework for AI-native networks," in *2025 IEEE 11th International Conference on Network Softwarization (NetSoft)*, 2025, pp. 55–60.

[3] J. Naillon. (2024, Nov.) Accelerating the autonomous network journey - a pathway to enhanced capabilities and strategic advantage. TM Forum Inform. [Online]. Available: https://inform.tmforum.org/features-and-opinion/accelerating-the-autonomous-network-journey-a-pathway-to-enhanced-capabilities-and-strategi

[4] European Parliament and Council, "Regulation (EU) 2024/1689 of the European Parliament and of the Council of 13 June 2024 laying down harmonised rules on artificial intelligence (Artificial Intelligence Act) and amending certain Union legislative acts," Official Journal of the European Union, OJ L, 2024/1689, Jun. 2024, available online: https://eur-lex.europa.eu/eli/reg/2024/1689/oj.

[5] F. Doshi-Velez and B. Kim, "Towards a rigorous science of interpretable machine learning," 2017. [Online]. Available: https://arxiv.org/abs/1702.08608

[6] S. M. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions," in *Advances in Neural Information Processing Systems*. Curran Associates, Inc., 2017. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2017/hash/8a20a8621978632d76c43dfd28b67767-Abstract.html

[7] S. M. Lundberg, G. Erion, H. Chen, A. DeGrave, J. M. Prutkin, B. Nair, R. Katz, J. Himmelfarb, N. Bansal, and S.-I. Lee, "From local explanations to global understanding with explainable AI for trees," *Nature Machine Intelligence*, vol. 2, no. 1, pp. 56–67, 2020.

[8] M. T. Ribeiro, S. Singh, and C. Guestrin, "Why should i trust you? explaining the predictions of any classifier," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2016.

[9] M. Sundararajan, A. Taly, and Q. Yan, "Axiomatic attribution for deep networks," 2017. [Online]. Available: https://arxiv.org/abs/1703.01365

[10] S. Wachter, B. Mittelstadt, and C. Russell, "Counterfactual explanations without opening the black box: Automated decisions and the gdpr," 2018. [Online]. Available: https://arxiv.org/abs/1711.00399

[11] J. v. Brocke, A. Hevner, and A. Maedche, *Introduction to Design Science Research*, 09 2020, pp. 1–13.

[12] M. Bello, R. Amador, M.-M. García, J. D. Ser, P. Mesejo, and Ó. Cordón, "The level of strength of an explanation: A quantitative evaluation technique for post-hoc XAI methods," *Pattern Recognition*, vol. 161, p. 111221, 2025.

[13] C.-K. Yeh, C.-Y. Hsieh, A. S. Suggala, D. I. Inouye, and P. Ravikumar, "On the (in)fidelity and sensitivity for explanations," 2019. [Online]. Available: https://arxiv.org/abs/1901.09392

[14] D. Alvarez-Melis and T. S. Jaakkola, "On the robustness of interpretability methods," 2018. [Online]. Available: https://arxiv.org/abs/1806.08049

[15] M. Nauta, J. Trienes, S. Pathak, E. Nguyen, M. Peters, Y. Schmitt, J. Schlötterer, M. van Keulen, and C. Seifert, "From anecdotal evidence to quantitative evaluation methods: A systematic review on evaluating explainable ai," *ACM Computing Surveys*, vol. 55, no. 13s, pp. 1–42, Jul. 2023. [Online]. Available: http://dx.doi.org/10.1145/3583558

[16] A. Clemm, L. Ciavaglia, L. Z. Granville, and J. Tantsura, "Intent-Based Networking – Concepts and Definitions," RFC 9315, Internet Engineering Task Force (IETF), Oct. 2022. [Online]. Available: https://www.rfc-editor.org/info/rfc9315

[17] A. Leivadeas and M. Falkner, "A survey on intent-based networking," *IEEE Communications Surveys and Tutorials*, vol. 25, no. 1, pp. 625–655, 2023, publisher Copyright: © 1998-2012 IEEE.

[18] ETSI, "Zero-touch network and service management (ZSM); reference architecture," ETSI GS ZSM 002 V1.1.1, 2019. [Online]. Available: https://www.etsi.org/deliver/etsi_gs/ZSM/001_099/002/01.01.01_60/gs_ZSM002v010101p.pdf

[19] 3GPP, "NR; overall description; stage-2 (release 17)," 3rd Generation Partnership Project, Technical Specification TS 38.300 V17.4.0, 2022.

[20] ——, "Study on scenarios and requirements for next generation access technologies (release 14)," 3rd Generation Partnership Project, Technical Report TR 38.913 V14.3.0, 2017.

[21] ——, "Management and Orchestration; Intent-Driven Management Services for Mobile Networks (Release 18)," 3rd Generation Partnership Project, Technical Specification TS 28.312 V18.4.0, 2024. [Online]. Available: https://www.3gpp.org/DynaReport/28312.htm

[22] B. Brik, H. Chergui, L. Zanzi, F. Devoti, A. Ksentini, M. S. Siddiqui, X. Costa-Pérez, and C. Verikoukis, "Explainable ai in 6g o-ran: A tutorial and survey on architecture, use cases, challenges, and future research," 2024. [Online]. Available: https://arxiv.org/abs/2307.00319

[23] A. Barredo Arrieta, N. Díaz-Rodríguez, J. Del Ser, A. Bennetot, S. Tabik, A. Barbado, S. García, S. Gil-López, D. Molina, R. Benjamins, R. Chatila, and F. Herrera, "Explainable artificial intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI," *Information Fusion*, vol. 58, pp. 82–115, 2020.

[24] M. T. Ribeiro, S. Singh, and C. Guestrin, ""why should I trust you?": Explaining the predictions of any classifier," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2016, pp. 1135–1144.

[25] R. K. Mothilal, D. Mahajan, C. Tan, and A. Sharma, "Towards unifying feature attribution and counterfactual explanations: Different means to the same end," in *Proceedings of the 2021 AAAI/ACM Conference on AI, Ethics, and Society*, 2021, pp. 652–663.

[26] S. Chari *et al.*, "Explanation ontology: A general-purpose, semantic representation for supporting user-centered explanations," *Semantic Web*, vol. 15, no. 4, pp. 959–989, 2024.

[27] L. S. Shapley, "A value for n-person games," *Contributions to the Theory of Games*, vol. 2, pp. 307–317, 1953.

[28] G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, and T.-Y. Liu, "LightGBM: A highly efficient gradient boosting decision tree," in *Advances in Neural Information Processing Systems*, vol. 30. Curran Associates, Inc., 2017.

[29] T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2016, pp. 785–794.

[30] L. Prokhorenkova, G. Gusev, A. Vorobev, A. V. Dorogush, and A. Gulin, "CatBoost: Unbiased boosting with categorical features," in *Advances in Neural Information Processing Systems*, vol. 31. Curran Associates, Inc., 2018.

[31] L. Grinsztajn, E. Oyallon, and G. Varoquaux, "Why do tree-based models still outperform deep learning on typical tabular data?" in *Advances in Neural Information Processing Systems (NeurIPS), Datasets and Benchmarks Track*, vol. 35, 2022.

[32] 3GPP, "System Architecture for the 5G System (5GS); Stage 2 (Release 17)," 3rd Generation Partnership Project, Technical Specification TS 23.501 V17.5.0, 2022. [Online]. Available: https://www.3gpp.org/DynaReport/23501.htm

[33] L. Rieger and L. K. Hansen, "Aggregating explanation methods for stable and robust explainability," 2020. [Online]. Available: https://arxiv.org/abs/1903.00519

[34] S. Krishna, T. Han, A. Gu, J. Pombra, S. Jabbari, S. Wu, and H. Lakkaraju, "The disagreement problem in explainable machine learning: A practitioner's perspective," 2022. [Online]. Available: https://arxiv.org/abs/2202.01602

[35] O. Mitrut, G. Moise, A. Moldoveanu, F. Moldoveanu, M. Leordeanu, and L. Petrescu, "Clarity in complexity: How aggregating explanations resolves the disagreement problem," *Artificial Intelligence Review*, vol. 57, no. 12, p. 338, 2024.

[36] U. Bhatt, A. Weller, and J. M. F. Moura, "Evaluating and aggregating feature-based model explanations," 2020. [Online]. Available: https://arxiv.org/abs/2005.00631

[37] I. Vascotto, A. Rodriguez, A. Bonaita, and L. Bortolussi, "Beyond single-model xai: aggregating multi-model explanations for enhanced trustworthiness," 2025. [Online]. Available: https://arxiv.org/abs/2510.11164

[38] S. Chatterjee, E. R. Colombo, and M. M. Raimundo, "Multi-criteria rank-based aggregation for explainable ai," 2025. [Online]. Available: https://arxiv.org/abs/2505.24612

[39] T. R. Henderson, S. Roy, S. Floyd, and G. F. Riley, "ns-3 project goals," in *Proceeding from the 2006 Workshop on ns-2: The IP Network Simulator*, 2006.

[40] N. Patriciello, S. Lagén, B. Bojovic, and L. Giupponi, "An E2E simulator for 5G NR networks," *Simulation Modelling Practice and Theory*, vol. 96, p. 101933, 2019.

A Complete Per-Domain Feature Sets

This appendix lists the complete feature sets used to train the per-domain and end-to-end models, referenced from Section 5.1.3.

Table 6: Complete per-domain feature sets used for model training. Type codes: R = Raw simulation parameter; D = Derived/engineered; L = Log-transform; S = Shared parameter carried with domain prefix in E2E. `run_id` is excluded as it is an identifier, not a predictive feature.

Domain	Type	Feature
RAN (22 features)		
RAN	R	num_ue
RAN	R	bandwidth_mhz
RAN	R	numerology
RAN	R	center_freq_ghz
RAN	R	cell_radius_m
RAN	R	ue_speed_mps
RAN	R	tx_power_dbm
RAN	R	packet_size_bytes
RAN	R	num_antennas
RAN	R	mimo_gain_db
RAN	R	snr_estimate_db
RAN	D	offered_load_mbps
RAN	D	bw_per_ue_hz
RAN	D	load_per_ue_mbps
RAN	D	ue_density_per_km2
RAN	D	doppler_spread_hz
RAN	D	ran_util_ratio
RAN	D	ran_queue_factor
RAN	L	log_num_ue
RAN	L	log_bandwidth_mhz
RAN	L	log_ue_density
RAN	L	log_doppler_hz
Core (20 features)		
Core	R	upf_rate_mbps
Core	R	upf_proc_ms
Core	R	cpu_utilization
Core	R	num_sessions
Core	R	core_loss_pct
Core	R	num_ue
Core	R	packet_size_bytes
Core	R	num_upf_cores
Core	D	offered_load_mbps
Core	D	upf_util_ratio
Core	D	cpu_queue_factor
Core	D	session_load_mbps

Continued on next page

Table 6 *(continued)*

Domain	Type	Feature
Core	D	upf_util_sq
Core	D	core_overload
Core	D	upf_queue_factor
Core	D	eff_upf_rate_mbps
Core	D	sessions_per_ue
Core	D	load_per_session
Core	L	log_upf_rate
Core	L	log_upf_proc_ms
Transport (21 features)		
Trans.	R	bh_rate_mbps
Trans.	R	link_distance_km
Trans.	R	num_hops
Trans.	R	hop_proc_ms
Trans.	R	backhaul_loss_pct
Trans.	R	num_ue
Trans.	R	packet_size_bytes
Trans.	R	tcp_cw_pkts
Trans.	D	offered_load_mbps
Trans.	D	fiber_delay_ms
Trans.	D	hop_total_delay_ms
Trans.	D	bh_util_ratio
Trans.	D	serial_delay_ms
Trans.	D	bdp_mbits
Trans.	D	bh_util_sq
Trans.	D	bh_overload
Trans.	D	bh_queue_factor
Trans.	D	eff_bh_rate_mbps
Trans.	D	retx_delay_ms
Trans.	L	log_bh_rate
Trans.	L	log_link_distance
E2E (63 features)		
E2E	R	num_ue
E2E	R	bandwidth_mhz
E2E	R	numerology
E2E	R	center_freq_ghz
E2E	R	cell_radius_m
E2E	R	ue_speed_mps
E2E	R	tx_power_dbm
E2E	R	packet_size_bytes
E2E	R	num_antennas
E2E	R	mimo_gain_db
E2E	R	snr_estimate_db
E2E	D	offered_load_mbps

Continued on next page

Table 6 *(continued)*

Domain	Type	Feature
E2E	D	bw_per_ue_hz
E2E	D	load_per_ue_mbps
E2E	D	ue_density_per_km2
E2E	D	doppler_spread_hz
E2E	D	ran_util_ratio
E2E	D	ran_queue_factor
E2E	L	log_num_ue
E2E	L	log_bandwidth_mhz
E2E	L	log_ue_density
E2E	L	log_doppler_hz
E2E	R	bh_rate_mbps
E2E	R	link_distance_km
E2E	R	num_hops
E2E	R	hop_proc_ms
E2E	R	backhaul_loss_pct
E2E	S	bh_num_ue
E2E	S	bh_packet_size_bytes
E2E	S	bh_offered_load_mbps
E2E	R	tcp_cw_pkts
E2E	D	fiber_delay_ms
E2E	D	hop_total_delay_ms
E2E	D	bh_util_ratio
E2E	D	serial_delay_ms
E2E	D	bdp_mbits
E2E	D	bh_util_sq
E2E	D	bh_overload
E2E	D	bh_queue_factor
E2E	D	eff_bh_rate_mbps
E2E	D	retx_delay_ms
E2E	L	log_bh_rate
E2E	L	log_link_distance
E2E	R	upf_rate_mbps
E2E	R	upf_proc_ms
E2E	R	cpu_utilization
E2E	R	num_sessions
E2E	R	core_loss_pct
E2E	S	core_num_ue
E2E	S	core_packet_size_bytes
E2E	S	core_offered_load_mbps
E2E	R	num_upf_cores
E2E	D	upf_util_ratio
E2E	D	cpu_queue_factor
E2E	D	session_load_mbps
E2E	D	upf_util_sq

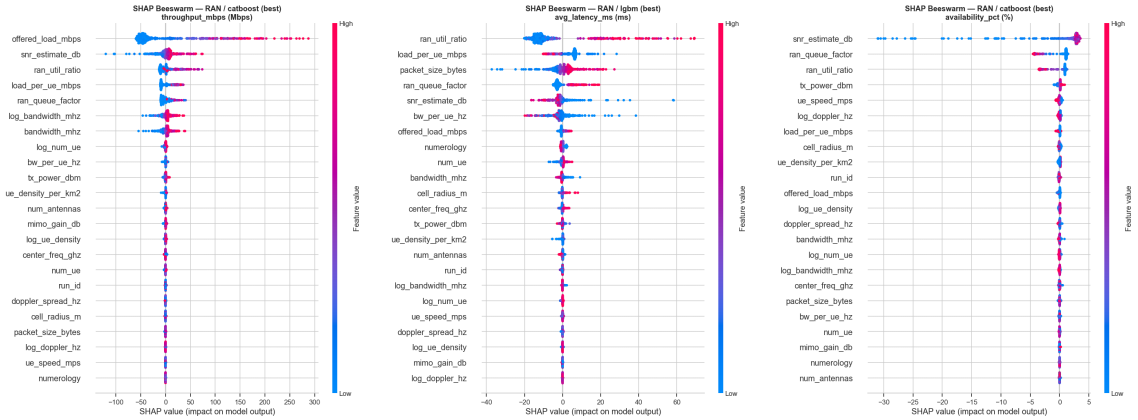
Continued on next page

Table 6 *(continued)*

Domain	Type	Feature
E2E	D	core_overload
E2E	D	upf_queue_factor
E2E	D	eff_upf_rate_mbps
E2E	D	sessions_per_ue
E2E	D	load_per_session
E2E	L	log_upf_rate
E2E	L	log_upf_proc_ms

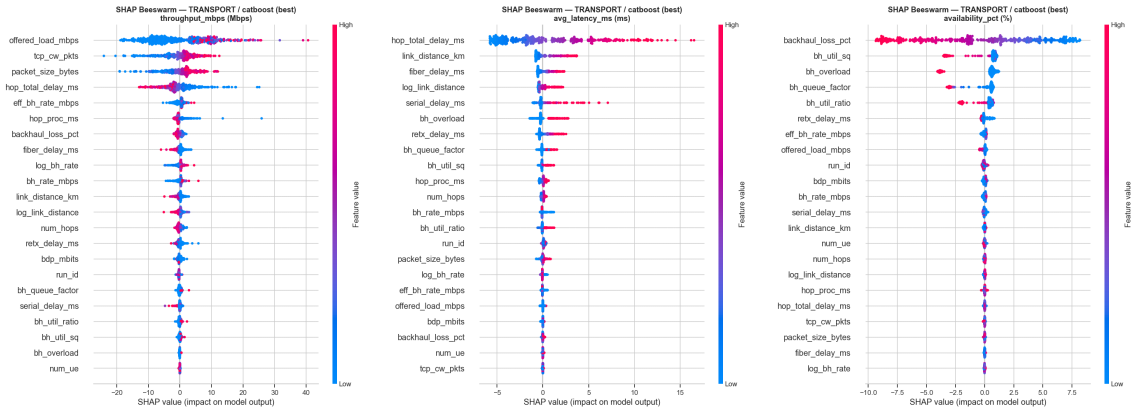
B Per-(Domain, KPI) SHAP Attribution Plots

This appendix collects the raw per-(domain, KPI) SHAP beeswarm plots and the cross-model-family comparison plots referenced from Section 7.3. Beeswarm plots show the distribution of SHAP values across the test split, with features ranked top-to-bottom by mean absolute SHAP value and instances colour-coded by feature value (red high, blue low).



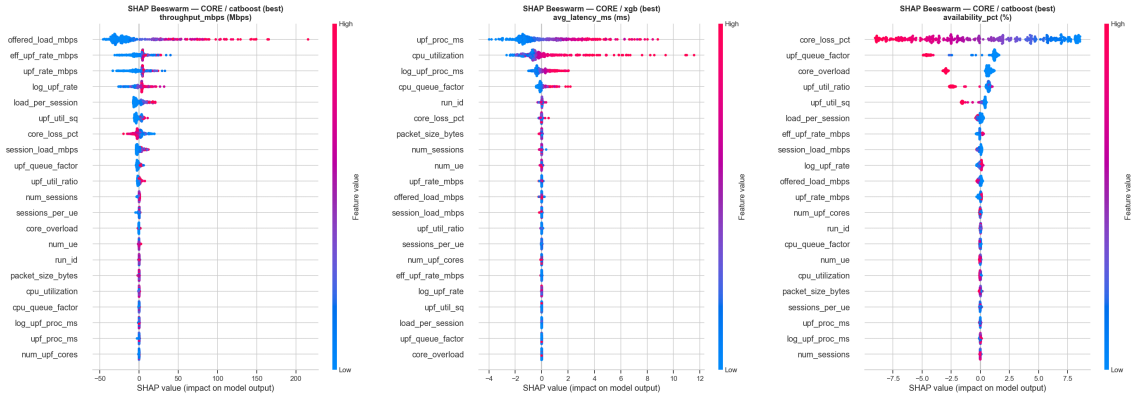
(a) Throughput (b) Latency (c) Availability

Figure 7: SHAP beeswarm plots for the best RAN model per KPI.



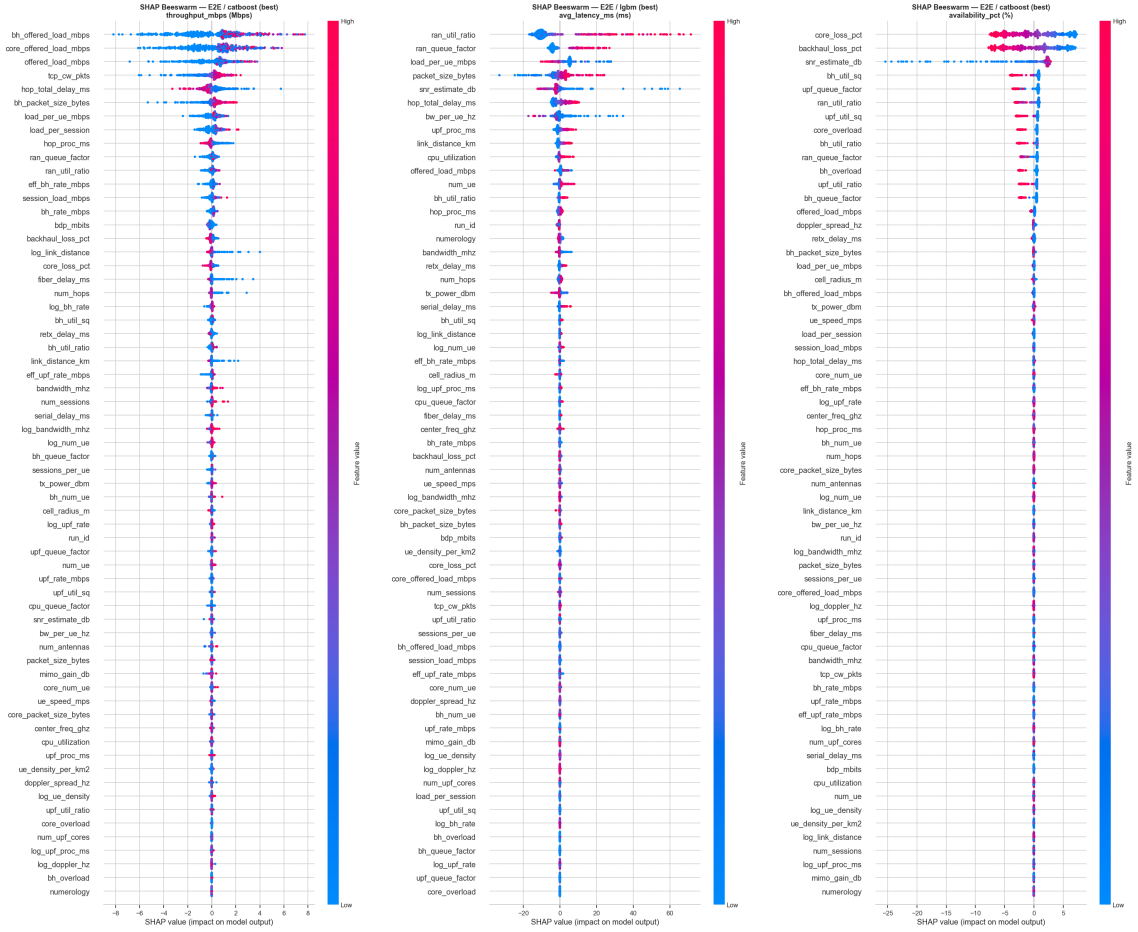
(a) Throughput (b) Latency (c) Availability

Figure 8: SHAP beeswarm plots for the best Transport model per KPI.



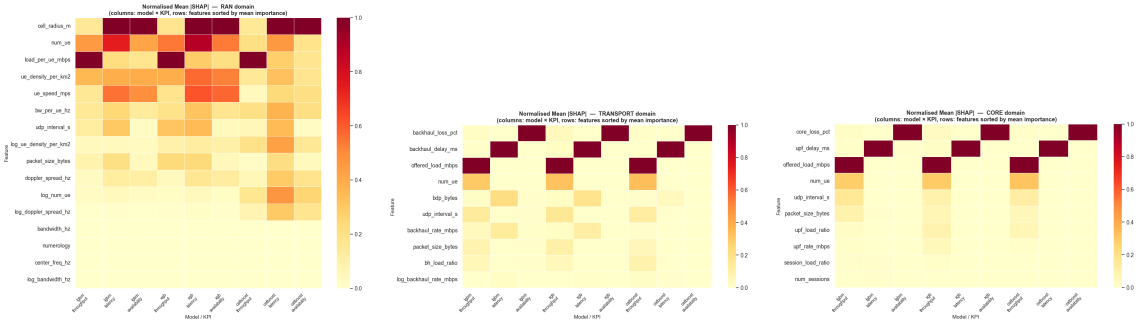
(a) Throughput (b) Latency (c) Availability

Figure 9: SHAP beeswarm plots for the best Core model per KPI.



(a) Throughput (b) Latency (c) Availability

Figure 10: SHAP beeswarm plots for the three best-selected E2E models (CatBoost for throughput and latency, LightGBM for availability). The E2E feature set is the union over all domains, allowing the model to capture cross-domain interactions inaccessible to the per-domain models.



(a) RAN (b) Transport (c) Core

Figure 11: SHAP feature-importance comparison across LightGBM, XGBoost, and CatBoost per domain. The top-ranked features agree across model families, supporting the use of the best-selected model's attributions as the per-(domain, KPI) explanation.