ФЕДЕРАЛЬНОЕ АГЕНТСТВО

ПО ТЕХНИЧЕСКОМУ РЕГУЛИРОВАНИЮ И МЕТРОЛОГИИ

П Р Е Д В А Р И Т Е Л Ь Н Ы Й
Н А Ц И О Н А Л Ь Н Ы Й
С Т А Н Д А Р Т
Р О С С И Й С К О Й
Ф Е Д Е Р А Ц И И

ПНСТ ___– 202_

Искусственный интеллект

АРХИТЕКТУРНАЯ СТРУКТУРА ДЛЯ МАШИННОГО ОБУЧЕНИЯ В БУДУЩИХ СЕТЯХ, ВКЛЮЧАЯ IMT-2020

Настоящий проект стандарта не подлежит применению до его утверждения

Москва

Российский институт стандартизации

202_

Предисловие

1 РАЗРАБОТАН Научно-образовательным центром компетенций в области цифровой экономики Федерального государственного бюджетного образовательного учреждения высшего образования «Московский государственный университет имени М.В.Ломоносова» (МГУ имени М.В.Ломоносова) и Обществом с ограниченной ответственностью «Институт развития информационного общества» (ИРИО)

2 ВНЕСЕН Техническим комитетом по стандартизации ТК 164 «Искусственный интеллект»

3 УТВЕРЖДЕН И ВВЕДЕН В ДЕЙСТВИЕ Приказом Федерального агентства по техническому регулированию и метрологии от 202_ г. №- ст

4 Настоящий стандарт разработан с учетом основных нормативных положений международного документа ITU-T Y.3172 (2019) «Архитектурная структура для машинного обучения в будущих сетях, включая IMT-2020» (Recommendation ITU-T Y.3172 (2019) «Architectural framework for machine learning in future networks including IMT-2020», MOD)

Правила применения настоящего стандарта и проведения его мониторинга установлены в ГОСТР 1.16—2011 (разделы 5 и 6).

Федеральное агентство по техническому регулированию и

метрологии собирает сведения о практическом применении настоящего стандарта. Данные сведения, а также замечания и предложения по содержанию стандарта можно направить не позднее чем за 4 мес до истечения срока его действия, разработчику настоящего стандарта по адресу: 119991, Российская Федерация, Москва, Ленинские горы, д. 1 и в Федеральное агентство по техническому регулированию и метрологии по адресу: 123112 Москва, Пресненская набережная, д. 10, стр. 2.

В случае отмены настоящего стандарта соответствующая информация будет опубликована в ежемесячном информационном указателе «Национальные стандарты» и также будет размещена на официальном сайте Федерального агентства по техническому регулированию и метрологии в сети Интернет (www.rst.gov.ru)

© Оформление. ФГБУ «Институт стандартизации», 202_

Настоящий стандарт не может быть полностью или частично воспроизведен, тиражирован и распространен в качестве официального издания без разрешения Федерального агентства по техническому регулированию и метрологии

Содержание

1 Область применения .....

2 Термины и определения.....

3 Сокращения .....

4 Соглашения по терминологии .....

5 Обзор.....

6 Высокоуровневые архитектурные требования .....

7 Структура высокоуровневой архитектуры .....

8 Требования обеспечения безопасности .....

Приложение А (справочное) Реализация высокоуровневой архитектуры в сетях с наложенным МО, зависящих от конкретной технологии .....

Приложение Б (справочное) Соотнесение архитектурных компонентов и поддерживающих аспектов с требованиями раздела 7 .....

Библиография .....

Введение

Настоящий стандарт определяет архитектурную структуру для машинного обучения в будущих сетях, включая IMT-2020. Представлен набор архитектурных требований и конкретных архитектурных компонентов, необходимых для удовлетворения этих требований. Эти компоненты включают, в частности, конвейер машинного обучения, а также функции управления и оркестровки машинного обучения, но не ограничиваются ими. Также описана интеграция этих компонентов в будущие сети, включая IMT-2020, и рекомендации по применению данной архитектурной структуры в различных сетях с наложенным машинным обучением, зависящих от конкретной технологии.

П Р Е Д В А Р И Т Е Л Ь Н Ы Й Н А Ц И О Н А Л Ь Н Ы Й С Т А Н Д А Р Т
Р О С С И Й С К О Й Ф Е Д Е Р А Ц И И

Искусственный интеллект

АРХИТЕКТУРНАЯ СТРУКТУРА ДЛЯ МАШИННОГО ОБУЧЕНИЯ В БУДУЩИХ СЕТЯХ, ВКЛЮЧАЯ IMT-2020

Artificial intelligence. Architectural framework for machine learning in future networks including IMT-2020

Срок действия – с __-____-____
до __-____-____

1 Область применения

Настоящий стандарт определяет архитектурную структуру для машинного обучения в будущих сетях, включая IMT-2020.

Представлен набор архитектурных требований, который, в свою очередь, приводит к разработке конкретных архитектурных компонентов, необходимых для удовлетворения этих требований. В настоящем стандарте также описывается архитектурная структура для интеграции таких компонентов в будущие сети, включая IMT-2020, и руководство по применению этой архитектурной структуры в различных сетях с наложенным машинным обучением, зависящих от конкретной технологии.

2 Термины и определения

В настоящем стандарте применены следующие термины с соответствующими определениями:

2.1 **конвейер машинного обучения** (machine learning pipeline): Набор логических узлов, каждый из которых обладает определенными функциональными возможностями, которые могут быть объединены для формирования приложения машинного обучения в телекоммуникационной сети.

Примечание – Узлы конвейера машинного обучения – это объекты, которые управляются стандартным образом и могут быть размещены в различных сетевых функциях.

2.2

машинное обучение (machine learning): Процесс оптимизации параметров модели с помощью вычислительных методов таким образом, чтобы поведение модели отражало данные и/или опыт.

[[1], пункт 3.3.5]

2.3

модель машинного обучения (machine learning model): Математическая конструкция, генерирующая логический вывод или прогноз на основе входных данных и/или информации.

Примечание – Модель машинного обучения является результатом обучения на основе алгоритма машинного обучения.

[1], пункт 3.3.7]

2.4 **наложенное машинное обучение** (machine learning overlay): Слабосвязанная модель развертывания функций машинного обучения, для которых стандартизованы процессы интеграции и управления с сетевыми функциями.

Примечание – Наложенное машинное обучение направлено на

минимизацию взаимозависимостей между функциями машинного обучения и сетевыми функциями благодаря использованию стандартных интерфейсов, обеспечивая тем самым параллельное развитие и тех, и других.

2.5 **оркестратор функций машинного обучения** (machine learning function orchestrator; MLFO): Логический узел с функциями управления и оркестровки узлов в конвейерах машинного обучения.

2.6 **полигон машинного обучения** (machine learning sandbox): Среда, в которой можно обучать, тестировать модели машинного обучения и оценивать их влияние на сеть.

Примечание – Полигон машинного обучения предназначен для предотвращения воздействия приложения машинного обучения на сеть или для ограничения использования определенных функций машинного обучения.

2.7

сетевая функция (network function): В контексте IMT-2020 — функция обработки в сети.

Примечания

1 Сетевые функции включают в себя функциональные возможности сетевых узлов, например, управление сеансом, управление мобильностью и транспортные функции, функциональное поведение и интерфейсы которых определены, но не ограничиваются ими.

2 Сетевые функции могут быть реализованы на выделенном оборудовании или в виде виртуализированных программных функций.

3 Сетевые функции не рассматриваются как ресурсы, скорее любые сетевые функции могут быть реализованы с использованием ресурсов.

[[2], пункт 3.2.8]

2.8 **сеть с наложенным машинным обучением** (machine learning underlay network): Телекоммуникационная сеть и связанные с ней сетевые функции, которые взаимодействуют с соответствующими функциями наложенного машинного обучения.

Примечание – Сеть IMT-2020 является примером сети с наложенным машинным обучением.

3 Сокращения

В настоящем стандарте применены следующие сокращения:

M — модель (в конвейере МО) (model in ML pipeline);

MO — машинное обучение (machine learning);

AF — функция приложения (application function);

AN — сеть доступа (access network);

API — интерфейс программирования приложений (application programming interface);

AR/VR — дополненная реальность/виртуальная реальность (augmented reality/virtual reality);

C — сборщик (в конвейере МО) (collector in ML pipeline);

CN — базовая сеть (core network);

D — распределитель (в конвейере МО) (distributor in ML pipeline);

EMS — система управления элементами (element management system);

FMC — конвергенция стационарной и мобильной связи (fixed mobile convergence);

GPU — графический процессор (graphic processor unit);

IoT — интернет вещей (internet of things);

KPI — ключевой показатель эффективности (key performance indicator);

mIoT — массивный интернет вещей (massive Internet of things);

MEC периферийные вычисления мультисервисного доступа (multi-access edge computing);

MLFO — оркестратор функций машинного обучения (machine learning function orchestrator);

MnS — служба управления (management service);

MPP — прогнозирование закономерностей мобильности (mobility pattern prediction);

NF — сетевая функция (network function);

NOP — сетевой оператор (network operator);

OAM — эксплуатация, администрирование и техническое обслуживание (operations, administration and maintenance);

P — политика (в конвейере МО) (policy in ML pipeline);

PP — предобработка (в конвейере МО) (preprocessor in ML pipeline);

QoS — качество обслуживания (quality of service);

RCA — анализ первопричин (root cause analysis);

RRC — управление радиоресурсами (radio resource control);

SBA — сервис-ориентированная архитектура (service-based architecture);

SMF — функция управления сеансом (session management function)

SON — самооптимизирующаяся сеть (self-optimizing network);

SRC — источник (source);

UE — пользовательское оборудование (user equipment);

UPF — функция плоскости пользователя (user plane function);

V2X — система обмена информацией между транспортным средством и его окружением (vehicle-to-everything);

VoLTE — голосовая связь в сетях LTE (Voice over Long-Term Evolution).

4 Соглашения по терминологии

В настоящем стандарте приняты следующие соглашения.

Ключевые слова «требуется, чтобы» означают требование, которое должно строго соблюдаться и отклонение от которого не допускается, если будет сделано заявление о соответствии настоящему стандарту.

Ключевое слово «рекомендуется» означает требование, которое рекомендуется, но не является абсолютно необходимым. Таким образом это требование не является обязательным для заявления о соответствии настоящему стандарту.

Ключевое слово «опционально» указывает на необязательное требование, которое допустимо, но не предполагает какого-либо свойства рекомендации. Этот термин не подразумевает, что реализация поставщика должна предоставлять такую возможность и эта функция может быть дополнительно включена сетевым оператором/поставщиком услуг. Скорее это означает, что поставщик может дополнительно предоставить функцию и по-прежнему заявлять о соответствии спецификации.

В основной части настоящего стандарта и приложениях к нему иногда встречаются такие слова, как «должен», «не должен», «следует» и «может», и в этом случае их следует интерпретировать соответственно как «требуется», «запрещено», «рекомендуется» и «опционально».

Появление таких фраз или ключевых слов в приложении или в материалах, явно помеченных как информативные, следует интерпретировать как не имеющее нормативного характера.

Конвейер МО – В настоящем стандарте символ, показанный на рисунке 1, обозначает подмножество (включая собственное подмножество) узлов в конвейере МО. Когда данный символ используется в рисунке, он обозначает подмножество узлов конвейера МО, которые явно не показаны на этом рисунке.

Рисунок 1 — Символ, используемый для обозначения подмножества узлов в конвейере МО

Точки выхода и входа сервиса – В настоящем стандарте узел и его точки выхода и входа сервиса обозначаются символом на рисунке 2. Точка выхода сервиса отображается зеленым прямоугольником в верхней части узла, а точка входа сервиса соответствует красному прямоугольнику в верхней части узла.

Рисунок 2 — Символ, используемый для обозначения узла с его точками выхода и входа сервиса

5 Обзор

Машинное обучение (МО) позволяет научить вычислительные системы получать знания из данных без необходимости явного программирования для реализации сложных задач, таких как обнаружение характеристик или прогнозирование поведения. Поскольку МО становится важной технической тенденцией в отрасли, сетевые операторы и другие заинтересованные стороны ищут экономически эффективные способы включения МО в будущие сети, включая IMT-2020.

Хотя преимущества такой интеграции обсуждались во многих случаях использования (например, устранение неполадок в сети, прогнозирование сетевого трафика, настройка оптимизации трафика, аудит информационной безопасности [3], [4]), такая интеграция сталкивается со множеством проблем. Проблемами являются:

А) гетерогенный характер функций МО и уникальные характеристики будущих коммуникационных технологий предъявляют разнообразный набор требований к интеграции;

Б) дорожные карты развития этих функций МО и сетей связи не согласованы;

В) стоимость интеграции с точки зрения влияния на архитектуру является важным фактором;

Г) разные механизмы управления функциональными возможностями машинного обучения и сетевыми функциями [2] нарушат управление операциями сетей связи.

Для решения этих проблем предназначена архитектурная структура для интеграции МО с будущими сетями, включая IMT-2020.

Основываясь на высокоуровневых архитектурных требованиях такая структура сначала предоставляет общий словарь и номенклатуру для функций МО и их взаимосвязей с сетями связи. Эта структура

определяет контрольные точки для будущих сетей, включая IMT-2020, которые обеспечивают слабосвязанную интеграцию с функциями машинного обучения. Механизмы управления МО также описаны в архитектурной структуре с учетом существующих принципов управления, которые были определены в сетях IMT-2020.

Эта структура обеспечивает стандартный метод интеграции функций МО в будущих сетях, включая IMT-2020.

6 Высокоуровневые архитектурные требования

В этом разделе представлены высокоуровневые требования к проектированию архитектурной структуры из раздела 7.

Данные высокоуровневые архитектурные требования классифицируются следующим образом:

– средства корреляции данных между уровнями и гетерогенными технологиями: этот набор требований адресован проблеме А) из раздела 5, входящие в него требования представлены в пункте 6.1;

Примечание – Уровни могут соответствовать различным сегментам сети, где контрольные точки между ними определены в соответствующих спецификациях сетевой архитектуры. Например, [5] определяет контрольную точку RP-an между сетью доступа (AN) и базовой сетью (CN), так что AN и CN могут соответствовать разным уровням сети IMT-2020.

– средства развертывания: этот набор требований адресован проблеме Б) из раздела 5, входящие в него требования представлены в пункте 6.2;

– требования, относящиеся к интерфейсам между архитектурными компонентами: этот набор требований адресован

проблемам Б) и В) из раздела 5, входящие в него требования представлены в пункте 6.3;

– требования, связанные с декларативными спецификациями, используемыми для спецификации приложений МО: этот набор требований адресован проблемам В) и Г) из раздела 5, входящие в него требования представлены в пункте 6.4.

– требования, связанные с управлением архитектурными компонентами: этот набор требований адресован проблеме Д) из раздела 5, входящие в него требования представлены в пункте 6.5.

6.1 Средства корреляции

В таблице 6.1 представлены требования к средствам корреляции данных между уровнями и гетерогенными технологиями.

Таблица 6.1 – Высокоуровневые требования – Средства корреляции

REQ-ML-COR-001	Рекомендуется использовать архитектуру МО для корреляции данных, поступающих из нескольких источников
Описание	В будущих сетях источники данных могут быть гетерогенными, интегрированными с различными сетевыми функциями (NF) и могут генерировать данные в разных форматах. Эти разнообразные «перспективы» могут дать ценную информацию при корреляционном анализе. Необходимы архитектурные компоненты, позволяющие функциям МО собирать и сопоставлять данные из различных источников в сети
Примечания	Примечание 1 – Например, анализ данных пользовательского оборудования (UE), сети доступа (AN), базовой сети (CN) и функции приложения (AF) необходим для прогнозирования потенциальных проблем, связанных с качеством обслуживания (QoS) в сквозных пользовательских потоках.

Продолжение таблицы 6.1

	Примечание 2 – Другими примерами таких источников данных являются функциональные возможности самооптимизирующейся сети (SON), которые отслеживают и коррелируют сетевые аварийные сигналы и ключевые показатели эффективности (KPI) [6], а затем предпринимают соответствующие действия для устранения аварийных сигналов или улучшения качества KPI сети или дают рекомендации по проектированию сети без вмешательства человека
REQ-ML-COR-002	а) Архитектура МО необходима для поддержки многочисленных технологий будущих сетей, включая IMT-2020, для обеспечения сквозного взаимодействия с пользователем. б) Рекомендуется, чтобы архитектура МО взаимодействовала с внешними функциональными объектами, не относящимися к IMT-2020, для обеспечения сквозного взаимодействия с пользователем
Описание	В будущих сетях будет сосуществовать множество технологий, например, лицензионные и нелицензионные беспроводные технологии, технологии конвергенции фиксированной мобильной связи (FMC), унаследованные и будущие технологии. Появление сегментации сети [7] является одним из примеров, в котором важны вертикальные технологии (сети, адаптированные для предоставления гибких решений для различных рыночных сценариев) и их интеграция в будущие сети. Взаимодействие архитектуры МО с такими функциональными объектами поможет в достижении KPI, как, например, указано в [6]. Таким образом, важно, чтобы архитектура МО была способна интегрироваться с несколькими базовыми технологиями и даже поддерживать функции приложений

Продолжение таблицы 6.1

Примечания	Примечание 1 – Система обмена информацией между транспортным средством и его окружением (V2X) является примером вертикального приложения, которое может получить выгоду от поддержки сегментации сети. Примечание 2 – Различные технологии сетей связи (например, 3G, 4G, 5G) могут рассматриваться как примеры базовых технологий. Приложения для организации развлечений в автомобиле, обработки данных с дронов, развлекательные приложения с использованием дополненной/виртуальной реальности (AR/VR) являются примерами функций приложений. Примечание 3 – Примерами сетевых функций, которые не управляются напрямую сетевым оператором (NOP), являются датчики, силовые цепи и различные модули интернета вещей (IoT). В некоторых случаях данные таких сетевых функций используются для управления и мониторинга как сетевых функций, развернутых в сети, так и внешних функциональных объектов сети. Эти данные также могут затем использоваться сетевым оператором (NOP) в различных типах оптимизации сетевых параметров для достижения увеличения покрытия, пропускной способности и качества
REQ-ML-COR-003	Архитектура МО необходима для поддержки распределенной реализации функций машинного обучения на нескольких уровнях
Описание	Распределенная реализация помогает использовать данные с разных уровней для обогащения (при необходимости) локально доступных данных с одного уровня. Таким образом, функциональные возможности МО могут быть многоуровневыми. Независимые экземпляры функций МО могут создаваться на нескольких уровнях

Окончание таблицы 6.1

Примечания	Примечание 1 – В некоторых вариантах использования источник данных может размещаться в сети доступа (AN), а предварительно обработанные данные использоваться моделью МО, которая находится в базовой сети (CN) [8]. Примечание 2 – Функциональные возможности МО могут использоваться для прогнозирования шаблонов мобильной связи (MPP) или конфигурации сегмента сети [2] с использованием входных данных из сети на нескольких уровнях (от источников в AN или CN)

6.2 Средства развертывания

В таблице 6.2 представлены требования к средствам развертывания функций МО.

Таблица 6.2 – Высокоуровневые требования – Средства развертывания

REQ-ML-DEP-001	Рекомендуется использовать архитектуру МО для определения точек взаимодействия между функциями МО и сетевыми функциями (NF) сетей с наложенным МО, зависящих от конкретной технологии, вне зависимости от функций приложения МО
Описание	Спецификация приложений МО в будущих сетях будет связана с существующими (или новыми) сетевыми службами или сетевыми функциями. На основе спецификации приложений МО определяются размещение и характеристики функций МО. Источником данных и целью результатов МО являются точки тесной интеграции с сетевыми функциями (NF) сети с наложенным МО, зависящими от конкретной технологии. Помимо этого, функции МО могут быть общими и не иметь тесной интеграции с сетевыми функциями NF, зависящими от конкретной технологии. Требуются четко определенные контрольные точки между функциями МО и сетевыми функциями NF сети с наложенным МО, зависящей от

Продолжение таблицы 6.2

	конкретной технологии
Примечания	Примечание 1 – Результатом МО могут быть политики или конфигурации, применяемые в сети, а целью результатов МО могут быть функции в сети для применения результатов МО. Такое применение результатов МО может контролироваться политикой сетевого оператора. Примечание 2 – В некоторых вариантах использования источник данных и классификация трафика (на основе результатов МО) могут быть размещены в плоскости пользователя сети, например функция плоскости пользователя (UPF) [5]. Их можно рассматривать как точки тесной интеграции между функциями МО и сетью с наложенным МО (например, сетью доступа AN или базовой сетью CN). Другая функциональность МО (например, модель МО) не имеет таких зависимостей интерфейса от сетей с наложенным МО
REQ-ML-DEP-002	Архитектура МО необходима для поддержки раздельного или комбинированного развертывания функций МО в различных сетевых функциях NF сетей с наложенным МО
Описание	В будущих сетях функции управления и оркестровки [7] будут соответствующим образом оптимизировать расположение и производительность сетевых функций (NF). Чтобы перенести эти преимущества на приложения МО, аналогичные оптимизации следует применить и к функциям МО. Более того, ограничения, применимые к функциям МО, могут быть уникальными
Примечания	Примечание 1 – Для машинного обучения может потребоваться графический процессор (GPU) и изолированная среда для исключения влияния на другие функции сети. Примечание 2 – Например, в зависимости от времени задержки, доступности данных и других особенностей приложений МО, функциональные возможности МО могут

Продолжение таблицы 6.2

	иметь источник данных и проводить обучение в CN или AN
REQ-ML-DEP-003	Архитектура МО необходима для поддержки гибкого размещения функций МО (в координации с функциями управления и оркестровки [2], [9]) в сетевых функциях сетей с наложенным МО
Описание	Гибкое размещение функций МО может основываться, среди прочего, на спецификациях приложений МО. Функции управления и оркестровки используют как спецификации приложений МО, так и условия в сети для реализации этого требования. При выделении ресурсов для функций МО необходимо учитывать различные ограничения (например, ограничения ресурсов NF, ограничения задержки специфические для конкретного приложения МО, доступность данных, зависящих от конкретного приложения МО, возможности преобразования данных, ограничения производительности, возможности обучения и характеристики модели). Архитектура МО должна обеспечивать возможность гибкого размещения функций МО в сети, наиболее оптимального для производительности приложений МО и на основе ограничений, определенных в декларативных спецификациях приложений МО. Ограничения для онлайнового обучения и прогнозирования для приложений МО в реальном времени, которые зафиксированы в спецификации, являются входными данными для размещения функций МО в сети, которые могут обеспечить оптимальную производительность для конкретного варианта использования
Примечания	Примечание 1 – Если модель МО включает нейронную сеть, то желательно ее разместить в системе из графических процессоров.

Окончание таблицы 6.2

	Примечание 2 – В зависимости от требований приложений МО функциональные возможности МО, обеспечивающие чувствительные к задержке краткосрочные прогнозы, могут размещаться ближе к периферии. На размещение также может влиять фактор доступности данных. Это можно скоординировать с разделением функций МО, упомянутым в REQ-ML-DEP-002. Примечание 3 – В некоторых вариантах использования классификация данных плоскости пользователя может выполняться с использованием моделей МО. Поскольку такая классификация является приложением, чувствительным к задержке, модель МО может размещаться в транспортной сети, а обучение выполняться в базовой сети CN
REQ-ML-DEP-004	Архитектуре МО необходимо поддерживать подключение и отключение новых источников данных или целевых объектов конфигурации в работающую среду МО
Описание	Некоторые продвинутые приложения в сетях будущего, например, массивный интернет вещей (mIoT), требуют обработки неструктурированных данных из огромного количества источников данных, которые могут быть готовы к использованию по принципу «включай и работай». Одним из таких вариантов использования является анализ зарегистрированных данных для обнаружения аномалий в сетях
Примечания	Примечание 1 – Масштабирование функций МО в зависимости от типа и объема входящих данных является примером обработки новых источников данных. Примечание 2 – Конфигурация функций МО в сети должна обеспечивать подключение и отключение новых источников данных или целевых конфигураций на основе метаданных об источниках данных

6.3 Связь с интерфейсом

В таблице 6.3 приведены требования, связанные с интерфейсом.

Таблица 6.3 – Требования высокого уровня, связанные с интерфейсом

REQ-ML-INT-001	Рекомендуется, чтобы архитектура МО поддерживала интерфейс для передачи обученных моделей между функциями МО нескольких уровней
Описание	Обучение моделей имеет определенные специфические потребности, например, доступность определенных типов процессоров, доступность определенных видов обучающих данных. После завершения обучения модель МО должна быть отправлена в сеть с наложенным МО, зависящую от конкретной технологии, и в которой размещена модель МО. Обучение модели может проводиться отдельно от эксплуатируемой сети. Таким образом, отправка обученных моделей на несколько уровней является важным требованием
Примечания	Примечание 1 – UE, AN, CN можно рассматривать как примеры сетевых функций сетей с наложенным МО, зависящих от конкретной технологии, в которых может размещаться обученная модель. Примечание 2 – CN и AN можно рассматривать как примеры нескольких уровней. Примечание 3 – В зависимости от доступности данных, обучение может проводиться в CN или AN, а сама обученная модель (например, классификатор) размещается в транспортных сетях
REQ-ML-INT-002	Архитектура МО должна поддерживать интерфейс передачи данных для обучения или тестирования моделей между функциями МО нескольких уровней
Описание	Определенные уровни с доступными данными могут не иметь возможности для обучения. В таких случаях может возникнуть необходимость в отправке данных для обучения или

Продолжение таблицы 6.3

	тестирования на уровни, имеющие такие возможности
Примечания	Примечание 1 – Сети доступа AN можно рассматривать как примеры сетей с ограниченными ресурсами, тогда как базовые сети CN можно рассматривать как уровень, на котором может быть доступен ресурс для масштабирования. Примечание 2 – Предварительная обработка данных может быть выполнена в транспортной сети, которые потом отправляются для обучения в модель МО, расположенную в базовой сети CN.
REQ-ML-INT-003	а) При определении интерфейса с сетями с наложенным МО в архитектуре МО рекомендуется использовать существующие стандартные протоколы везде, где это возможно, с необходимыми расширениями всюду, где это необходимо. б) Рекомендуется, чтобы архитектура МО поддерживала специальные интерфейсы или интерфейсы программирования приложений (API) для взаимодействия с зависящими от конкретной технологии сетевыми функциями NF, для получения данных от них или для их конфигурирования. в) Рекомендуется, чтобы архитектура МО поддерживала логические интерфейсы между функциональными возможностями МО, которые могут быть размещены на нескольких уровнях, а реализация таких логических интерфейсов осуществлялась в соответствии со сценариями развертывания
Описание	Источникам данных и целям результатов МО могут потребоваться специальные интерфейсы или интерфейсы программирования приложений API с сетями с наложенным МО для извлечения данных или применения конфигураций. В некоторых вариантах использования может потребоваться тесная связь на этапе интеграции между источником и целью функциональных возможностей МО и сетевыми функциями NF. В некоторых случаях может потребоваться расширение таких

Окончание таблицы 6.3

Описание	интерфейсов для достижения функциональных возможностей МО в конкретном варианте использования. Функциональные возможности МО могут использовать интерфейсы, предоставляемые сетью с наложенным МО, в качестве источника данных или цели конфигураций. В этом смысле такие сетевые API могут выступать как реализация интерфейса к источнику и цели функциональных возможностей МО
Примечания	Примечание 1 – В отношении пункта а) выше, некоторые интерфейсы могут быть реализованы путем повторного использования существующих протоколов (таких как управление радиоресурсами (RRC) [5], MEC [10], служба управления (MnS) [11]). Примечание 2 – В отношении пункта б) выше, например, источник, работающий с пользовательским оборудованием (UE), может использовать специальные API для извлечения данных из клиентского приложения обеспечивающего голосовую связь в сетях LTE (VoLTE) [12]. Примечание 3 – В отношении пункта в) выше, например, в случае, когда источник работает в сети доступа (AN), но требует измерений от пользовательского оборудования (UE), AN должна настроить пользовательское оборудование для этих измерений с помощью протокола RRC
REQ-ML-INT-004	В архитектуре МО рекомендуется поддерживать совместное использование данных между функциональными возможностями МО на основе механизмов распределенного межуровневого совместного использования данных
Описание	Межуровневое совместное использования данных необходимо для принятия коррелированных решений МО в будущих сетях
Примечания	Примечание – Такие концепции, как озера данных, появляются в будущих облаках и также могут быть использованы в облаках сетевых операторов

6.4 Декларативные спецификации

В таблице 6.4 приведены требования, касающиеся декларативных спецификаций.

Таблица 6.4 – Требования высокого уровня – Декларативная спецификация

REQ-ML-SPEC-001	Архитектуре МО необходимо поддерживать стандартный метод представления приложений МО, который может быть преобразован в функциональные возможности МО в сетевых функциях NF сетей с наложенным МО, зависящих от конкретных технологий
Описание	Характерной чертой будущих сетей является автоматизация с использованием декларативной спецификации и соответствующего преобразования в необходимые конфигурации. При распространении данного метода на МО необходимо поддерживать декларативную спецификацию приложений МО и соответствующие преобразования в конфигурации функциональных возможностей МО
Примечания	Интерпретация декларативной спецификации позволяет конфигурировать функциональные возможности МО в сети путем преобразования спецификации в конфигурацию, которая может быть размещена сетевыми функциями
REQ-ML-SPEC-002	Архитектуре МО необходимо поддерживать приложения МО для спецификации источников данных, репозиториев моделей МО, целевых результатов моделей МО и ограничений на сетевые ресурсы. Архитектуре МО необходимо поддерживать требования к ограничениям по времени, предъявляемым к приложениям МО
Описание	Разделение между технологически независимой частью приложения МО и его развертыванием, зависящим от конкретной технологии, учитывается во время разработки сервисов будущих сетей. Декларативные спецификации для

Продолжение таблицы 6.4

	приложений МО обеспечивают это разделение. Различные приложения МО имеют различные временные ограничения. Эти ограничения формируют важные входные данные для функций управления и оркестровки при определении местоположения, объединения в цепочку и мониторинга функциональных возможностей МО
Примечания	Примечание 1 – В отношении вышеупомянутого пункта а), например, описание, созданное на метаязыке, может отражать требования оператора сети к приложению МО. Функции управления и оркестровки могут преобразовать их в конфигурацию, которая может быть реализована в сети. Примечание 2 – В отношении вышеупомянутого пункта б), в самом узком масштабе применение МО в сетевых функциях формирования луча, планирования, адаптации канала связи будет иметь критерии задержки порядка микросекунд, тогда как для сетевых функций транспортных и базовых сетей критерии задержки составляют несколько миллисекунд. Наименее требовательными с точки зрения задержки являются функции на уровне управления, например, обнаружение аномалий и «дыр» в покрытии, для которых допустима задержка на минуты, часы или дни
REQ-ML-SPEC-003	Архитектура МО необходима для поддержки гибкого разделения функциональных возможностей МО на основе спецификаций приложений МО
Описание	Спецификация приложения МО является важным вкладом в развертывание МО в будущих сетях, включая IMT-2020. Но сетевые возможности могут изменяться (аппаратное обеспечение может быть добавлено или удалено), сетевые функции NF могут быть запланированы или динамически (повторно) настроены с помощью функций управления и оркестровки. Эти динамические изменения могут потребовать изменения в разделении и местоположении функциональных

Продолжение таблицы 6.4

	возможностей МО (например, может быть принято решение о совместном размещении определенных функций МО на основе изменений в пропускной способности канала, или может быть принято решение о добавлении нового источника данных на основе применения функций управления и оркестровки). Таким образом, необходима комбинация (а) исходных данных из спецификации, (б) требований к функциям МО, которые допускают раздельное и комбинированное развертыванию, и (в) координации с базовой функцией управления и оркестровки
Примечания	Примечание – Новый источник данных может быть реализован на основе решений о масштабировании в сети
REQ-ML-SPEC-004	Архитектура МО необходима для поддержки спецификаций приложений МО третьими сторонами для определения источников данных, репозиториев моделей МО, целевых результатов моделей МО и ограничений на сетевые ресурсы
Описание	Сторонние [2] поставщики услуг могут предлагать инновационные услуги поверх будущих сетей. Данные услуги могут включать в себя новые алгоритмы МО. Сотрудничество между сторонними поставщиками и сетевыми операторами может привести к появлению новых источников данных или возможностей для агрегирования. Декларативный процесс в архитектуре должен расширять возможности для включения сторонних поставщиков, способных включать эти функциональные возможности в спецификацию, чтобы конечные пользователи могли воспользоваться инновационными услугами, предлагаемыми сторонними поставщиками

Окончание таблицы 6.4

Примечания	Примечание 1 – В некоторых случаях модели МО могут предоставляться в виде приложений сторонних производителей. Примечание 2 – Приложение для смартфона, которое взаимодействует с датчиками на пользовательском оборудовании, является примером стороннего источника данных. Примечание 3 – Сторонний поставщик услуг передачи голосовых сообщений с использованием интернет-протокола (Voice over IP) хочет оптимизировать качество звонков по сети, запустив приложение МО, конфигурирующее параметры сети. Спецификация такого приложения МО является примером спецификации стороннего поставщика

6.5 Управление функциональными возможностями МО

В таблице 6.5 приведены требования, касающиеся управления функциональными возможностями МО.

Таблица 6.5 – Требования высокого уровня – Управление функциональными возможностями МО

REQ-ML-MNG-001	Архитектуре МО необходимо поддерживать выбор модели МО во время настройки функциональных возможностей МО
Описание	Достижения в области машинного обучения предполагают, что в будущих сетях будут использоваться модели МО с различными характеристиками (например, с использованием различных методов оптимизации и весовых коэффициентов), подходящие для различных проблемных пространств и характеристик данных. В будущих сетях новые источники данных могут добавляться динамически. Чтобы распространить приложения МО на новые и гетерогенные источники данных, выбор модели МО должен

Продолжение таблицы 6.5

	осуществляться динамически, на основе данных, предоставленных источниками
Примечания	Примечание – Подключение новых сетевых функций (например, новой функции в плоскости пользователя UPF) к эксплуатируемой сети может служить примером динамического подключения новых источников данных
REQ-ML-MNG-002	Архитектуре МО необходимо поддерживать обучение и обновление модели МО, предотвращая при этом воздействие на сеть
Описание	При обучении модели МО учитывается несколько факторов: использование специального оборудования для повышения скорости, доступность данных, оптимизация параметров, избежание предвзятости, распределение обучения и т.д. Кроме того, в будущих сетях следует избегать сбоев в обслуживании во время обучения модели и ее обновления
Примечания	–
REQ-ML-MNG-003	Архитектуре МО необходимо поддерживать возможности мониторинга сетевых операций на основе эффекта МО и обновления моделей МО и/или политик без ущерба для сети
Описание	Необходимо отслеживать эффекты воздействия МО на сеть. Различные KPI постоянно подвергаются измерению, поэтому влияние машинного обучения на них, а также на сами функции МО необходимо постоянно отслеживать и, при необходимости, корректировать. Функциональные возможности МО необходимо обучить для будущего распознавания и обработки подобных скорректированных сценариев
Примечания	Примечание – Примером процесса, в котором осуществляется мониторинг и обновление в сети, является непрерывное совершенствование рабочих процессов автоматизированного восстановления после сбоев. Анализ первопричин (RCA) процесса восстановления после сбоя

Продолжение таблицы 6.5

	предоставляется для настройки функций управления и оркестровки, а также оценивается эффект, производимый МО в сети, для последующей оптимизации самих функций МО
REQ-ML-MNG-004	Архитектуре МО необходимо поддерживать функциональность оркестровки для управления всеми функциональными возможностями МО в сети
Описание	Производительность функциональных возможностей МО в сети отслеживается, и когда производительность падает ниже заданного порога, функциональные возможности МО перенастраиваются для ее улучшения. Разнообразные источники данных (такие как базовая сеть CN, сеть доступа AN) приводят к тому, что могут применяться различные методы обучения, включая распределенное обучение. Сложные модели могут быть обучены с использованием разнообразных данных. Производительность таких моделей можно определить и сравнить в изолированной среде. Основываясь на таких сравнениях, операторы сетей могут затем выбрать соответствующие функциональные возможности МО (на основе внутренних политик) для конкретных приложений МО
Примечания	Примечание 1 – Оценка включает в себя оценку производительности сети наряду с производительностью алгоритмов МО. Примечание 2. Повторный выбор модели МО является примером перенастройки, выполненной для повышения производительности функциональных возможностей МО
REQ-ML-MNG-005	Архитектуре МО необходимо поддерживать гибкое объединение функциональных возможностей МО в цепочку, включая многоуровневую цепочку

Окончание таблицы 6.5

Описание	Требуется гибкое объединение функциональных возможностей МО в цепочку в зависимости от хостинга и позиционирования в разных сетевых функциях NF и на разных уровнях. Это делается для возможности использования гибридных или распределенных функций МО. Объединение функциональных возможностей МО в цепочку может быть использовано для создания комплексных функциональных возможностей МО. Функции управления и оркестровки предоставляют сетевым операторам (NOP) возможности для быстрого проектирования, разработки и развертывания сетевых сервисов в сетях с наложенным МО, зависящих от конкретных технологий. Аналогичным образом, функции МО в сети нуждаются в механизмах, включающих гибкое объединение в цепочку, чтобы соответствовать инновациям в сетях с наложенным МО. По мере того, как сетевые сервисы сетей с наложенным МО быстро развиваются и развертываются, то же самое происходит и с функциональными возможностями МО поверх них, благодаря использованию гибких методов объединения в цепочку. Это требование направлено на обеспечение функциональным возможностям МО способности адаптироваться к динамическому созданию сервисов и оркестровке в сетях с наложенным МО
Примечания	–

7 Структура высокоуровневой архитектуры

Структура высокоуровневой архитектуры, описанная в данном разделе, поддерживает требования, перечисленные в разделе 6.

В этом разделе приводится описание высокоуровневой архитектуры и ее компонентов, а также даны рекомендации по реализации

высокоуровневой архитектуры в различных сетях с наложенным МО, зависящих от конкретной технологии.

Такой подход к проектированию архитектуры обеспечивает возможность анализа как решений МО, инвариантных относительно сетей с наложенным МО, зависящих от конкретных технологий, так и специфичных вопросов интеграции МО в такие сети с наложенным МО. Пример реализации этой архитектурной структуры в сетях, зависящих от конкретной технологии, представлен в приложении А.

7.1 Высокоуровневые архитектурные компоненты

В данном разделе определены высокоуровневые архитектурные компоненты, которые являются существенными частями архитектурной структуры. Интеграция таких компонентов в архитектуру сети путем взаимодействия с сетевыми функциями NF, наряду с определением местоположения функциональных возможностей МО в такой сети, образует архитектурную структуру.

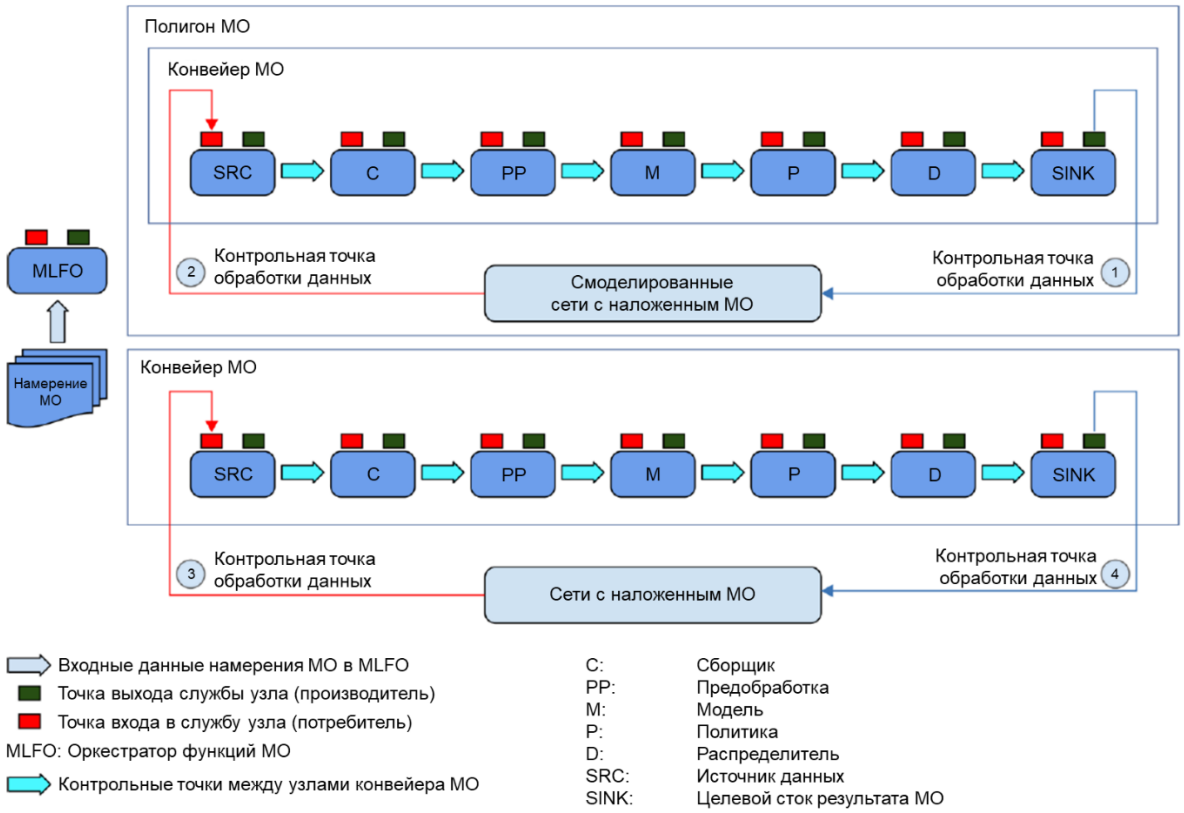

Рисунок 3 – Компоненты высокоуровневой архитектуры

Высокоуровневые архитектурные компоненты включают в себя следующее:

1) Конвейер машинного обучения

Как определено в разделе 2, конвейер машинного обучения – это набор логических узлов, каждый из которых обладает определенными функциональными возможностями, которые могут быть объединены для формирования приложения машинного обучения в телекоммуникационной сети.

Примечание 1 – Конвейер машинного обучения может быть развернут в моделируемых или эксплуатируемых сетях с наложенным МО.

Ниже описаны узлы конвейера МО:

– SRC (источник): этот узел является источником данных, которые могут быть использованы в качестве входных данных для конвейера МО;

Примечание 2 – Потенциальные узлы источника включают пользовательское оборудование (UE), функцию управления сеансом (SMF) [5] и функцию приложения (AF) [5].

– C (сборщик): этот узел отвечает за сбор данных из одного или нескольких источников;

Примечание 3 – Узел сборщика может иметь возможность настраивать узлы источников. Например, протокол управления радиоресурсами (RRC) [11] может использоваться для настройки пользовательского оборудования (UE), действующего в качестве узла источника. Узел сборщика может использовать протоколы операций, администрирования и обслуживания (OAM), специфичные для поставщика, для настройки SMF, действующего как узел источника. Такие конфигурации могут использоваться для управления характером данных, их детализацией и периодичностью при их генерации из источника.

– PP (предобработка): этот узел отвечает за очистку данных, агрегирование данных или выполнение любой другой предварительной обработки, необходимой для приведения данных в форму, подходящую для использования моделью МО;

– M (модель): это модель МО в форме, подходящей для использования в конвейере машинного обучения;

Примечание 4 – Примером является алгоритм МО, реализованный в программном обеспечении как сетевая функция NF [5].

– P (политика): этот узел позволяет применять политики к результатам работы модели из соответствующего узла;

Примечание 5 – Этот узел можно использовать, например, для минимизации воздействия, когда результаты машинного обучения применяются в эксплуатируемых сетях с наложенным МО. Оператор сети может ввести специальные правила для обеспечения работоспособности сети, например, серьезные обновления могут производиться только в ночное время или при низком трафике данных в сети.

– D (распределитель): этот узел отвечает за идентификацию целевых стоков (SINK) и распределение результатов из узла модели (M) по соответствующим узлам целевых стоков (SINK);

Примечание 6 – Данный узел может иметь возможность настраивать узлы целевых стоков. Например, протокол RRC может использоваться для настройки пользовательского оборудования (UE), действующего в качестве узла целевого стока (SINK).

– SINK (целевой сток): Этот узел является целью вывода МО, над которым он выполняет действие;

Примечание 7 – Например, пользовательское оборудование (UE), действующее в качестве узла целевого стока (SINK), может регулировать периодичность измерения канала на основе результатов МО.

Примечание 8 – «Точка выхода службы» узла – это точка, в которой можно получить доступ к службам, предоставляемым данным узлом (роль производителя). «Точка входа службы» узла – это точка, в которой могут быть использованы службы других узлов (роль потребителя).

2) Оркестратор функций машинного обучения

Как определено в разделе 2, оркестратор MLFO – это логический узел с функциями управления и оркестровки узлов в конвейерах МО на

основе намерений МО и/или динамических условий сети.

Примечание 9 – Оркестратор MLFO выбирает и повторно выбирает модель МО, основываясь, например, на ее производительности.

Примечание 10 – Размещение узлов конвейера МО, основанное на соответствующих сетевых возможностях и ограничениях приложения МО, является ответственностью оркестратора MLFO.

Примечание 11 – Оркестратор MLFO также предоставляет функциональность соединения узлов в цепочку для формирования конвейера МО. Например, соединение в цепочку может использоваться для связи узла источника SRC, реализованного в сети доступа, с узлом сборщика C и узлом предобработки PP, реализованными в базовой сети. Сама цепочка декларируется в спецификации приложения МО, а ее конкретная технологическая реализация в сети осуществляется оркестратором MLFO. Оркестратор MLFO определяет соединение узлов в цепочку с учетом ограничений (например, временных ограничений для прогнозирования).

Примечание 12 – На рисунке 4 показано намерение МО, чтобы подчеркнуть его важную роль в качестве входных данных для оркестратора MLFO. Намерение МО – это декларативное описание, которое используется для спецификации приложения МО. Намерение МО не определяет какие-либо сетевые функции со специфическими технологиями, которые будут использоваться в приложении МО, а обеспечивает основу для отображения приложений МО в различные сетевые функции со специфическими технологиями. Намерение МО для определения приложений МО может использовать метаязык, специфичный для машинного обучения.

3) Полигон машинного обучения

Полигон МО – это изолированная среда, которая позволяет размещать отдельные конвейеры МО для их обучения, тестирования и оценки перед развертыванием в эксплуатируемой сети. Для обучения или тестирования полигон МО может использовать данные, сгенерированные моделируемыми и/или эксплуатируемыми сетями с наложенным МО .

Примечание 13 – Для обучения, тестирования и оценки моделей МО на полигоне МО могут пригодиться такие методы как МО с учителем.

В дополнение к описанию вышеупомянутых высокоуровневых архитектурных компонентов следует обратить внимание на следующие вспомогательные архитектурные аспекты:

– архитектура, основанная на сервисах (SBA) [13], может использоваться для сопряжения функциональных возможностей МО и сетей с наложенным МО. Аналогично, для конвейера МО на полигоне данная архитектура может использоваться для сопряжения функциональных возможностей МО и моделируемых сетей с наложенным МО. Это обеспечивает единую контрольную точку как для наложенного МО для сетевых функций NF, так и для моделируемых сетей с наложенным МО. Архитектура SBA также может использоваться как между самими узлами конвейера МО, так и для управления функциональными возможностями МО с помощью оркестратора MLFO;

– контрольные точки оперирования данными определяются между конвейером МО и моделируемыми или эксплуатируемыми сетями с наложенным МО. Используя эти контрольные точки, любое воздействие на сети с наложенным МО локализуется в источнике данных и целевой конфигурации (в результате выполнения конвейера МО).

Примечание 14 – Расширения существующих протоколов могут использоваться в контрольных точках оперирования данными для минимизации воздействия архитектуры на сети с наложенным МО.

Примечание 15 – В случае когда сетевые функции в сетях с наложенным МО не совместимы с архитектурой SBA, в контрольных точках оперирования данными также следует использовать протоколы, не поддерживающие требования SBA,

Применительно к рисунку 3 стрелками 2 и 3 показаны пути к конвейеру МО для данных, сгенерированных из моделируемых сетей с наложенным МО и сетей с наложенным МО, соответственно. Стрелки 1 и

4 показывают пути из конвейера МО для настройки целевого объекта на основе результатов МО в моделируемых сетях с наложенным МО и сетях с наложенным МО, соответственно.

В приложении Б приведено соответствие архитектурных компонентов и вспомогательных архитектурных аспектов требованиям высокого уровня из раздела 6.

7.2 Высокоуровневая архитектура

Высокоуровневая архитектура, представленная на рисунке 4, основана на требованиях высокого уровня из раздела 6, и опирается на архитектурные компоненты, описанные в 7.1.

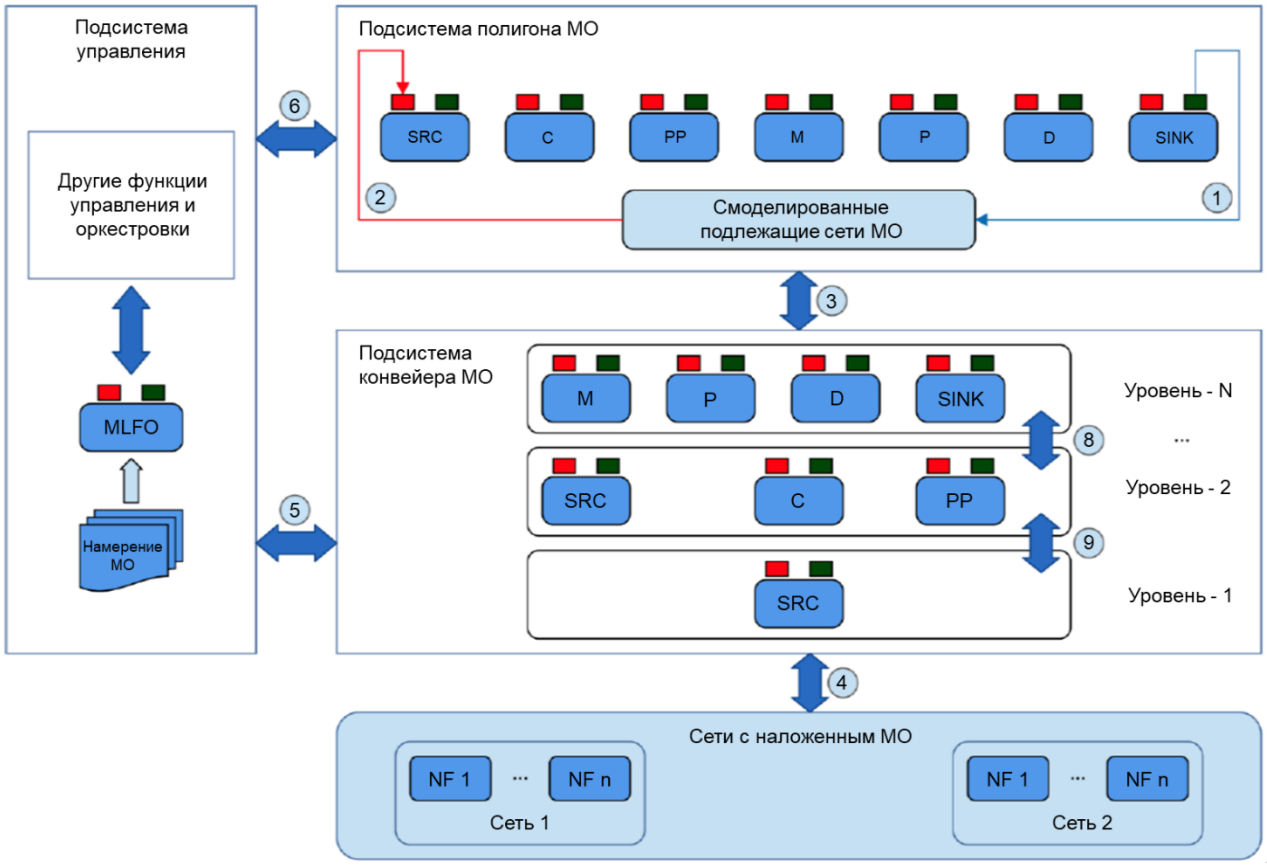

Рисунок 4 — Высокоуровневая архитектура

На рисунке 4 изображены следующие контрольные точки:

1, 2: Контрольные точки оперирования данными между моделируемыми сетями с наложенным МО и конвейером МО в подсистеме полигона МО.

3: Контрольная точка между подсистемой полигона МО и подсистемой конвейера МО.

4: Контрольная точка между подсистемой конвейера МО и сетями с наложенным МО.

Примечание 1 — Данная контрольная точка обозначает контрольные точки оперирования данными, представленные на рисунке 3 в виде стрелок 3 и 4.

5, 6: Контрольные точки между подсистемой управления, подсистемой конвейера МО и подсистемой полигона МО соответственно.

7: Контрольная точка между оркестратором функций машинного обучения (MLFO) и другими функциями управления и оркестровки подсистемы управления.

8, 9: Контрольные точки между узлами конвейера МО, расположенными на разных уровнях.

Ниже приведены три подсистемы высокоуровневой архитектуры, представленной на рисунке 4.

1) Подсистема управления

Данная подсистема включает в себя оркестратор MLFO и другие функции управления и оркестровки, например, определенные в [7].

Подсистема управления позволяет расширить механизмы управления и оркестровки узлов конвейера МО, используемые для будущих сетей, включая IMT-2020. Это обеспечивает единый подход в управлении функциональными возможностями МО и сетевыми функциями.

Оркестратор MLFO работает в координации с другими функциями

подсистемы управления с целью контроля узлов конвейера МО.

Примечание 2 — Контрольные точки между функциями подсистемы управления и сетевыми функциями NF в сетях с наложенным МО соответствуют контрольным точкам, определенным в соответствующих спецификациях, например, [7]. Данные контрольные точки выходят за рамки данного стандарта и потому не показаны на рисунке 4.

Примечание 3 — Взаимодействие между оркестратором MLFO и другими функциями подсистемы управления может осуществляться с использованием архитектуры SBA.

2) Подсистема конвейера машинного обучения

Конвейер МО представляет собой логический конвейер, который может быть наложен на существующие сетевые инфраструктуры. Службы оркестратора MLFO используются для реализации и настройки узлов. При интеграции наложения конвейера МО на сети с наложенным МО, зависящие от конкретной технологии, может потребоваться расширение или определение специальных протоколов и API.

Кроме того, необходимо отметить следующее:

– Архитектура SBA может использоваться для обеспечения связи как между сетевыми функциями NF и узлами конвейера МО, так и между самими узлами конвейера МО. Например, узел источника (SRC) предоставляет доступ к интерфейсам для потребления данных от сетевых функций (NF) и производства данных для узла коллектора (C). Другой пример — узел с целевым стоком (SINK) предоставляет доступ к интерфейсам для потребления результатов МО от узла с распределителем (D) и производит конфигурации для сетевых функций (NF), с которыми имеет связь;

Примечание 4 — Вследствие гетерогенности сетевых функций и сетей с

наложенным МО, архитектура SBA может не поддерживаться сетевыми функциями в сетях с наложенным МО. В таких случаях между узлами конвейера МО и сетевыми функциями используются протоколы и API, специфичные для таких сетей с наложенным МО.

– размещение и построение цепочек узлов конвейера МО контролируется оркестратором MLFO, на осуществление этого контроля могут повлиять такие факторы, как:

– входные данные от намерения МО к оркестратору MLFO, которые могут накладывать ограничения на размещение узлов конвейера МО;

Примечание 5 — Примером ограничений по расположению узла модели МО (М) является требование расположить узел М на сетевом вычислительном ресурсе, предоставляющем определенный тип возможностей для ускорения.

– обратная связь, получаемая оркестратором MLFO от функций управления и оркестровки, относящихся к сетям с наложенным МО, или от узлов конвейера МО, может предоставить входные данные для размещения и построения цепочек узлов конвейера МО;

Примечание 6 — Отвязка местоположения узлов конвейера МО от их функциональных возможностей, за исключением случаев ограничения производительности, достигается с помощью механизмов размещения и построения цепочек.

– развертывание конвейера МО в будущих сетях, включая IMT-2020, может охватывать разные уровни. В этом случае для обеспечения многоуровневого распределения конвейера МО используются

контрольные точки между узлами, расположенные на разных уровнях конвейера МО.

Примечание — Как показано на рисунке 4, конвейер МО может быть распределен по нескольким уровням, например, при конкретном развертывании конвейер может быть распределен между пользовательским оборудованием UE, сетью доступа AN и базовой сетью CN. В зависимости от конкретного приложения МО возможны различные способы распределения узлов конвейера МО.

3) Подсистема полигона машинного обучения

Подсистема полигона МО позволяет конвейерам МО адаптироваться к динамическим сетевым средам, таким как будущие сети, включая IMT-2020, в которых возможно изменение ряда условий (например, условия радиоинтерфейса, расположение пользовательского оборудования, возможности и ресурсы сети). Подсистема полигона МО включает в себя конвейер(ы) МО и моделируемые сети с наложенным МО и управляется оркестратором MLFO в соответствии со спецификациями намерения МО. Подсистема полигона МО позволяет сетевым операторам изучать влияние результатов МО перед их развертыванием в эксплуатируемых сетях с наложенным МО. Обратная связь от функций подсистемы управления поступает в подсистему полигона МО для того, чтобы конвейеры МО в подсистеме полигона МО могли адаптироваться к динамически меняющимся сетевым средам.

Необходимо отметить следующие моменты:

– контрольная точка между подсистемой конвейеров МО и подсистемой полигона МО позволяет конвейерам МО взаимодействовать с подсистемой полигона МО для обучения и обновления моделей МО;

– данные из сетей с наложенным МО и/или моделируемых сетей с наложенным МО могут использоваться для обучения моделей МО в подсистеме полигона МО;

– управление узлами конвейера МО в подсистеме полигона МО также контролируется оркестратором MLFO. Это позволяет MLFO обучать и выбирать модель(-и) МО для конкретного приложения МО.

7.3 Общие рекомендации по реализации высокоуровневой архитектуры

Общие рекомендации по реализации высокоуровневой архитектуры в различных сетях с наложенным МО, зависящих от конкретных технологий, сводятся к следующему:

– реализация узлов конвейера МО: приложение МО описывается с помощью намерения МО. Поток информации в приложении МО может быть представлен с помощью цепочки узлов в конвейере МО. Данные из различных узлов с источниками, например, поступающие из различных сетей с наложенным МО, должны быть собраны в узле со сборщиком и предварительно обработаны в узле с предобработкой до их передачи в узел с моделью МО. Затем результаты модели МО используются для применения в узле политики, которые будут реализованы в узле с целевым стоком. Приложение МО может быть создано путем реализации узлов конвейера МО с определенными ролями (например, источник SRC, сборщик C, целевой сток SINK) и привязки этих узлов к сетевым функциям в сети с наложенным МО, зависящей от конкретной технологии, исходя из соответствующих требований приложения МО и возможностей сетевых функций в сети с наложенным МО. Реализация выполняется оркестратором MLFO в координации с другими функциями управления и оркестровки.

– взаимодействие приложения МО с сетевыми функциями в сети с наложенным МО: для приложения МО существует две точки специфического взаимодействия с сетевыми функциями в сети с наложенным МО – узлы с источниками SRC и целевыми стоками SINK. Узлы с источниками могут иметь как основанный на архитектуре SBA интерфейс к сетевым функциям (производящим данные), так и не поддерживающий архитектуру SBA интерфейс к сетевым функциям, зависящим от конкретной технологии. Подобным образом узлы с целевыми стоками могут иметь основанный на архитектуре SBA интерфейс для связанных с ним сетевыми функциями NF, обеспечивающими исполнение выходных политик МО, либо не поддерживающий архитектуру SBA интерфейс к сетевым функциям NF, зависящим от конкретной технологии;

– управление конвейером МО: осуществляется оркестратором MLFO в координации с другими функциями управления и оркестровки;

– обучение и оценка моделей МО на полигоне МО: управляется оркестратором MLFO независимо от сетей с наложенным МО. Контрольная точка между подсистемой полигона МО и подсистемой конвейера МО используется для передачи обученных моделей МО, данных для обучения и обновлений моделей МО между подсистемой полигона МО и сетями с наложенным МО, которые взаимодействуют с реальным конвейером МО.

Приложение А содержит примеры применения вышеуказанных рекомендаций при реализации высокоуровневой архитектуры в сетях с наложенным МО, зависящих от конкретной технологии.

8 Требования обеспечения безопасности

Настоящий стандарт описывает архитектурную структуру МО,

которую предполагается применять в будущих сетях, включая IMT-2020: в этой связи следует применять общие требования и механизмы сетевой безопасности в сетях на базе интернет-протокола IP [14], [15].

Требуется предотвращение несанкционированного доступа и утечки данных из конвейера МО с использованием механизмов аутентификации и авторизации, защиты от внешних атак и т. д., независимо от того, являются действия злоумышленными или нет.

Приложение А
(справочное)

Реализация высокоуровневой архитектуры в сетях с наложенным МО, зависящих от конкретной технологии

На рисунке А.1 приведен пример реализации высокоуровневой архитектуры в сети IMT-2020 [5], [7].

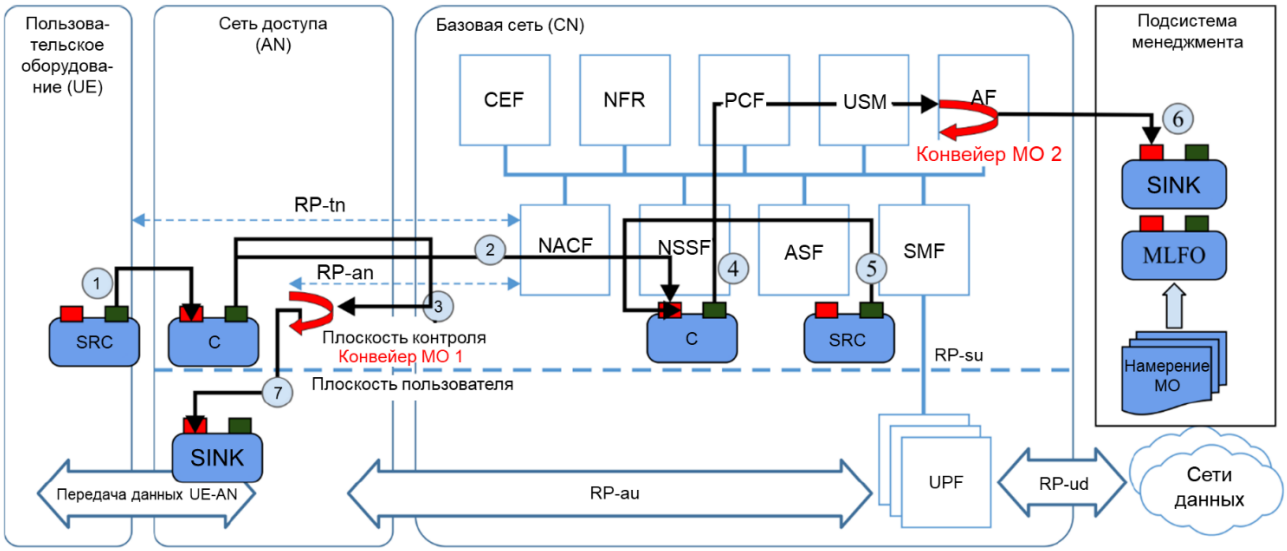

Рисунок А.1 — Пример реализации высокоуровневой архитектуры в сети IMT-2020

Данный пример реализации представлен следующим образом: конвейер МО демонстрирует расположение узлов конвейера МО в местах размещения узлов, например, в CN, AN, пользовательском оборудовании или функциях управления. Например, конвейер МО, представленный стрелками 1→2→4→конвейер МО 2, использует входные данные от пользовательского оборудования для прогнозирования в CN (например, приложения МО на основе МРР).

Примечание – Для простоты понимания подсистема полигона не показана на рисунке А.1, но ее функциональность применима и к рисунку А.1.

Далее описываются различные варианты реализации применительно к требованиям, указанным в разделе 6:

– экземпляр реализации, поддерживающий требования 6.1 и 6.2:

а) стрелки 5 → 4 → конвейер МО 2 → 6: данный конвейер МО использует входные данные от CN и, если необходимо, комбинацию входных данных от пользовательского оборудования, чтобы сделать прогнозы в CN и применить их к функциям управления. Такое применение результатов МО может, в свою очередь, повлиять на конфигурации различных уровней (например, на решения SON, принимаемые в CN, или на решения о распределении ресурсов, принимаемые в сети по замкнутому циклу);

б) стрелки 1 → 3 → конвейер МО 1 → 7: данный конвейер МО использует входные данные от пользовательского оборудования и размещает модель МО в AN для принятия решений, чувствительных к задержке, которые применяются в самой AN;

– экземпляр реализации, поддерживающий требования 6.3:

а) стрелка 1: возможна реализация с помощью RRC;

б) стрелки 2, 3, 4, 5, 7: возможна реализация в виде расширения контрольных точек в CN [5];

в) стрелка 6: возможна реализация путем повторного использования контрольных точек, определенных в [7];

– экземпляр реализации, поддерживающий требования 6.4:

а) пользовательское оборудование является устройством с ограниченными ресурсами, поэтому на пользовательском оборудовании задается только источник. Данное ограничение указывается в намерении МО;

– экземпляр реализации, поддерживающий требования пункта 6.5:

а) MLFO размещает сборщики в AN и CN на основе спецификаций приложений МО в намерении МО. Для приложений, чувствительных к задержкам в AN, используется конвейер МО 1. Конвейер МО 2 используется для приложений, устойчивых к задержкам. Объединение в цепочку осуществляется в соответствии с требованиями, указанными в намерении МО.

Приложение Б
(справочное)

Соотнесение архитектурных компонентов и поддерживающих аспектов с требованиями раздела 7

В таблице В.1 приведено соотнесение архитектурных компонентов и вспомогательных аспектов с требованиями раздела 6.

Таблица В.1 — Соотнесение архитектурных компонентов и вспомогательных аспектов с требованиями

Архитектурные компоненты и вспомогательные аспекты	Требования	Объяснение соотнесения
Конвейер МО	См. пункты 6.1 и 6.2	Конвейер МО предоставляет общую терминологию для МО в будущих сетях, включая IMT-2020. Определение узлов конвейера МО позволяет следить за эволюцией МО в отдельности от сетей с наложенным МО. Задание таких узлов является важной частью развертывания МО в различных сетях с наложенным МО. Функции этих узлов определяются независимо от их расположения в сети, что делает возможным гибкое размещение таких узлов в сети. Сегментация, слияние и объединение в цепочку узлов конвейера МО позволяет разворачивать сложные функции из этих базовых узлов

Продолжение таблицы В.1

Архитектурные компоненты и вспомогательные аспекты	Требования	Объяснение соотнесения
MLFO	См. пункты 6.2, 6.4 и 6.5	MLFO оркестрирует узлы конвейера МО и управляет ими. Он также отвечает за оптимальное размещение в сети и объединение узлов конвейера МО в цепочку. MLFO реализует эти функции в координации с функциями управления и оркестровки. Декларативные спецификации приложений МО поддерживаются MLFO путем преобразования их в развертывания, специфичные для сети с наложенным МО. Он также выбирает и перевыбирает модели, исходя из потребностей приложений МО и других ограничений, определенных в намерении МО
Полигон МО	См. пункт 6.5	Данная подсистема создает возможность обучения, оценки и мониторинга производительности моделей МО перед их развертыванием в реальной сети. Она взаимодействует с сетями с наложенным МО для передачи данных или обученных моделей. Кроме того, в ней используются моделируемые сети с наложенным МО, которые могут генерировать данные, необходимые для обучения моделей МО

Окончание таблицы В.1

Архитектурные компоненты и вспомогательные аспекты	Требования	Объяснение соотнесения
Намерение МО	См. пункт 6.4	Данный вспомогательный аспект архитектуры, написанный на метаязыке, дает определение приложению МО, его потребностям в плане ввода данных, вывода конфигурации, любых других ограничений или характеристик. Он используется в качестве входных данных для функциональных возможностей MLFO. Это исходные данные для MLFO для создания развертываемого, соответствующего требованиям конвейера МО в конкретных сетях с наложенным МО. Стандартизация данного архитектурного аспекта позволяет сторонним поставщикам решений интегрироваться с конвейером МО
SBA	См. пункт 6.3	Данный вспомогательный аспект архитектуры используется для взаимодействия между сетевыми функциями сети с наложенным МО и конвейером МО. Он обеспечивает необходимое слабосвязанное взаимодействие между наложенным МО и сетями с наложенным МО. Архитектура SBA также используется для взаимодействия между моделируемыми сетями с наложенным МО и конвейером МО. Она используется для обучения конвейера МО и настройки моделируемых сетей МО с помощью конвейера МО

Библиография

[1]	ИСО/МЭК 22989:2022	Информационные технологии. Искусственный интеллект. Концепции искусственного интеллекта и терминология
[2]	ITU-T Y.3100	Термины и определения для сетей IMT-2000
[3]	ITU-T Y.3650	Общие принципы сетевого взаимодействия на основе больших данных
[4]	ITU-T Y.3170	Требования к системам машинного обучения для обеспечения качества обслуживания в сети IMT-2020
[5]	ITU-T Y.3104	Архитектура сети IMT-2020
[6]	3GPP TS 28.554	Управление и оркестровка сетей 5G; ключевые показатели эффективности 5G (KPI) (Выпуск 15)
[7]	ITU-T Y.3111	Общие принципы управления сетью и оркестровки IMT-2020
[8]	ITU-T Y.3102	Общие принципы сети IMT-2020
[9]	ITU-T Y.3110	Требования к управлению сетью и оркестровке IMT-2020
[10]	ETSI MEC 003	Вычисления на мобильных периферийных устройствах (MEC); общие принципы и эталонная архитектура
[11]	3GPP TS 23.501	Системная архитектура для системы 5G (Выпуск 15)
[12]	GSMA IR.92	Профиль IMS для голосовой связи и SMS

[13]	ETSI TS 129 500	Система 5G; техническая реализация архитектуры на основе обслуживания; этап 3 (3GPP TS 29.500, версия 15.0.0, выпуск 15)
[14]	ITU-T Y.2701	Требования к безопасности для сетей последующих поколений версии 1
[15]	ITU-T Y.3101	Требования сети IMT-2020
[16]	ITU-T Q.5001	Требования к сигнализации и архитектура интеллектуальных периферийных вычислений
[17]	ITU-T Y.2322	Функциональная архитектура управления и оркестратора субъектов виртуализированных управляющих сетей в эволюции сетей последующих поколений
[18]	ITU-T Y.3105	Требования к отображению возможностей в сети IMT-2020
[19]	ITU GLOSSARY	Широкополосный доступ: Сокращения, аббревиатуры и отраслевая терминология https://www.itu.int/osg/spu/ni/broadband/glossary.html
[20]	ETSI NFV-IFA 014	Виртуализация сетевых функций (NFV), выпуск 2; управление и оркестровка; спецификация шаблонов сетевых сервисов
[21]	IEC	Белая книга МЭК по периферийным вычислительным средствам http://www.iec.ch/whitepaper/pdf/IEC_WP_Edge_Intelligence.pdf

УДК 004.8:006.354 ОКС 35.020

Ключевые слова: искусственный интеллект, высокоуровневая архитектура, IMT-2020, намерение, машинное обучение, оркестратор, наложенное машинное обучение, конвейер, требования, полигон, обучение, сеть с наложенным машинным обучением

Руководитель разработки и
исполнитель
Председатель совета директоров
Института развития
информационного общества Ю. Е. Хохлов