

Федеральное государственное автономное образовательное учреждение  
высшего образования «Московский физико-технический институт  
(национальный исследовательский университет)»

Физтех-школа аэрокосмических технологий

Кафедра Аэрофизики и летательных аппаратов

**Направление подготовки:** 09.03.01 Информатика и вычислительная техника  
(бакалавриат)

**Направленность (профиль) подготовки:** Компьютерное моделирование

**Форма обучения:** очная

## ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА

**«Алгоритм предиктивного анализа отказов системы видеоаналитики в  
режиме времени по данным от систем мониторинга»**

(бакалаврская работа)

**Студент:**

Боровец Николай Васильевич

---

*(подпись студента)*

**Научный руководитель:**

Гришин Никита Александрович,  
программист ПИШ РПИ

---

*(подпись научного руководителя)*

Жуковский

2025

## АННОТАЦИЯ

Выпускная квалификационная работа посвящена разработке метода предиктивного анализа задержек в конвейере видеоаналитики для мониторинга объектов критической инфраструктуры. **Цель работы** — создать алгоритм прогнозирования метрики *common\_event\_delay* с автоматическим обнаружением аномалий для предупреждения операторов о потенциальных сбоях. В работе применяются **методы исследования**, включающие анализ временных рядов Prometheus-метрик, сравнение архитектур ML-моделей (таких как трансформеры и градиентный бустинг), временную кросс-валидацию, развертывание в Docker и A/B-тестирование. В результате исследования разработан MLOps-конвейер с точностью прогнозирования, превышающей базовые методы, и временем отклика менее 1 секунды. Создана интеллектуальная система оповещений с адаптивными порогами. Проведена валидация разработанного решения на исторических данных объемом 90643 точки, собранных за 16 дней. **Практическая значимость** работы заключается в создании готового к использованию решения для предиктивного мониторинга видеосистем критической инфраструктуры с возможностью адаптации для применения в телекоммуникациях и промышленной автоматизации.

**Ключевые слова:** предиктивный анализ, задержки, видеоаналитика, Prometheus, временные ряды, аномалии.

## СОДЕРЖАНИЕ

АННОТАЦИЯ . . . . .	2
СОДЕРЖАНИЕ . . . . .	3
ВВЕДЕНИЕ . . . . .	4
1 Общие положения . . . . .	9
1.1 Архитектура системы видеоаналитики . . . . .	9
1.2 Постановка задачи . . . . .	10
2 Глава n . . . . .	15
2.1 Секция n . . . . .	15
3 Глава n . . . . .	16
3.1 Секция n . . . . .	16
4 Глава n . . . . .	17
4.1 Секция n . . . . .	17
ЗАКЛЮЧЕНИЕ . . . . .	18
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ . . . . .	19
Список сокращений и условных обозначений . . . . .	20

## ВВЕДЕНИЕ

### **Обоснование выбора темы и актуальность**

Современные системы видеоаналитики играют критически важную роль в обеспечении безопасности и мониторинга объектов критической инфраструктуры, включая аэродромы, железнодорожные станции, морские порты, промышленные предприятия и нефтеперерабатывающие комплексы. Эти системы обрабатывают огромные объемы видеоданных в режиме реального времени, что предъявляет высокие требования к производительности и надежности всего технологического конвейера.

С ростом масштабов развертывания и усложнением архитектуры видеоаналитических систем возрастает и сложность их мониторинга. Современные решения часто включают в себя многоуровневые конвейеры обработки, начиная от захвата видеопотоков с камер, их предварительной обработки, применения алгоритмов машинного обучения для детекции объектов и событий, передачи результатов через брокеры сообщений в бэкенд-системы и далее к конечным пользователям через веб-интерфейсы.

Повышение объемов данных и жестких требований к end-to-end-задержкам (от момента возникновения события на видео до его отображения оператору) делает необходимым переход от реактивного к предиктивному подходу в управлении производительностью. Традиционные методы мониторинга, основанные на статических пороговых значениях и алертах по факту превышения SLA, не способны предотвратить деградацию качества обслуживания до ее критических проявлений.

В данном контексте особую важность приобретает разработка интеллектуальных систем предиктивного анализа, способных на основе пото-

ковых метрик мониторинга (например, собираемых системой Prometheus) заблаговременно предсказывать потенциальные проблемы производительности и инициировать превентивные меры по их устранению.

### **Цель и задачи исследования**

*Цель работы:* разработать и внедрить комплексный метод предиктивного анализа задержек в конвейере видеоаналитики, способный прогнозировать критическую метрику *common\_event\_delay* с заданной точностью и автоматически детектировать аномальные паттерны в работе системы для предупреждения операторов о потенциальных сбоях до их фактического проявления.

Достижение поставленной цели требует решения комплекса взаимосвязанных *задач*:

1. Проведение аналитического обзора и систематизация современной литературы по предиктивному анализу временных рядов, машинному и глубокому обучению в контексте мониторинга и диагностики производительности систем реального времени.
2. Проведение глубокого анализа структуры и взаимных корреляций временных рядов метрик, собираемых на всех критических этапах видеоконвейера, включая выявление скрытых зависимостей между компонентами системы и идентификацию наиболее информативных признаков для прогнозирования.
3. Систематический обзор и сравнительный анализ современных методов прогнозирования временных рядов и обнаружения аномалий, включая классические статистические подходы, методы машинного обучения и глубокие нейронные сети, с оценкой их применимости к специфике видеоаналитических конвейеров.

4. Обоснованный выбор оптимальной архитектуры модели (трансформер, градиентный бустинг или их гибридная комбинация) с учетом требований к точности и скорости inference, а также определение необходимого объема обучающих данных и оптимальной периодичности переобучения модели.
5. Проектирование и реализация полноценного MLOps-конвейера, включающего автоматизированный feature-engineering, механизмы периодического дообучения модели на новых данных, высокопроизводительный inference-сервис и системы мониторинга качества предсказаний.
6. Всестороннее экспериментальное исследование точности и производительности разработанной модели на обширных исторических данных с использованием методов временной кросс-валидации и оценкой устойчивости к различным типам аномалий в данных.
7. Разработка и внедрение интеллектуальной системы оповещений с адаптивными порогами, а также формулирование практических рекомендаций по эксплуатации, настройке и масштабированию решения в производственной среде.

### **Методология и методы исследования**

Для достижения поставленной цели и решения сформулированных задач применяется комплексная методология, сочетающая теоретические исследования с практическими экспериментами:

1. Организация непрерывного сбора и интеллектуальной предобработки потоковых метрик из системы мониторинга Prometheus, включая очистку от выбросов, нормализацию, обработку пропущенных значений и синхронизацию временных рядов различных компонентов системы.

2. Разработка специализированного модуля построения многомерных временных рядов с интеллектуальной генерацией признаков, включая временные лаги различной глубины, скользящие статистические агрегаты, спектральные характеристики и высокоразмерные эмбединги для захвата сложных временных зависимостей.
3. Реализация и экспериментальное сравнение различных архитектур моделей (трансформеры с механизмом внимания, ансамбли градиентного бустинга LightGBM/CatBoost, гибридные нейро-символьные подходы) с применением строгих методов перекрёстной валидации по времени для обеспечения корректной оценки обобщающей способности.
4. Контейнеризация решения с использованием технологии Docker и проведение детальных измерений latency inference в условиях, максимально приближенных к производственным, включая тестирование под нагрузкой и оценку масштабируемости.
5. Организация и проведение А/В-тестирования в реальной производственной среде с использованием методов статистической оценки значимости результатов и анализа влияния на ключевые показатели эффективности системы.

### **Теоретическая и практическая значимость**

*Теоретическая значимость* работы заключается в расширении фундаментальных знаний о применимости и эффективности гибридных подходов к онлайн-прогнозированию сложных многомерных временных рядов в условиях высоких требований к задержкам и точности предсказаний. Исследование вносит вклад в теорию адаптивного машинного обучения для динамических систем реального времени и методологию проектирования отказоустойчивых MLOps-конвейеров.

*Практическая значимость* определяется разработкой готового к промышленному использованию решения для мониторинга и предупреждения отказов видеоконвейера с гарантированным соблюдением SLA по конечной метрике *common\_event\_delay*. Созданная система может быть адаптирована и масштабирована для применения в различных отраслях, где критична надежность систем обработки потоковых данных в реальном времени, включая телекоммуникации, финансовые технологии и промышленную автоматизацию.



## **1 Общие положения**

Данная глава посвящена формальной постановке задачи предиктивного анализа задержек в конвейере видеоаналитики и представлению архитектуры исследуемой системы. В рамках главы вводятся ключевые математические обозначения, определяются целевые метрики и ограничения, формулируются требования к разрабатываемому алгоритму. Особое внимание уделяется описанию структуры видеоконвейера и точек сбора телеметрических данных, которые лягут в основу построения прогностической модели.

### **1.1 Архитектура системы видеоаналитики**

Исследуемая система видеоаналитики представляет собой сложный многокомпонентный конвейер, предназначенный для обработки видеопотоков в режиме реального времени с применением алгоритмов машинного обучения для детекции событий и объектов. Архитектура системы строится по принципу микросервисной организации, что обеспечивает масштабируемость и отказоустойчивость, но одновременно усложняет задачи мониторинга и диагностики производительности.

Видеоконвейер включает следующие основные компоненты: модуль захвата видеопотока с IP-камер (получающий данные по протоколу RTSP), ML-pipeline для применения алгоритмов компьютерного зрения, брокер сообщений Apache Kafka для асинхронной передачи результатов обработки, бэкенд-сервисы для бизнес-логики и сохранения данных, а также WebSocket-клиенты для доставки уведомлений конечным пользователям. Каждый компонент генерирует множество метрик производительности, которые собираются централизованной системой мониторинга Prometheus.

Критической характеристикой системы является end-to-end-задержка, измеряемая как время от момента возникновения события в видеопотоке до его отображения на интерфейсе оператора. Данная метрика, обозначаемая как *common\_event\_delay*, напрямую влияет на эффективность работы операторов и качество принимаемых ими решений в критических ситуациях.

## 1.2 Постановка задачи

Для формальной постановки задачи прогнозирования введем необходимые математические обозначения и определения. Пусть  $T = \{t_1, t_2, \dots, t_n\}$  — упорядоченное множество временных меток наблюдений, соответствующих моментам сбора метрик из системы мониторинга с фиксированным интервалом дискретизации. Обозначим через  $d$  общее число различных метрик, одновременно собираемых системой мониторинга со всех компонентов видеоконвейера.

Для каждой временной метки  $t_i$  формируется  $d$ -мерный вектор наблюдений:

$$\mathbf{x}_i = [m_i^{(1)}, m_i^{(2)}, \dots, m_i^{(d)}] \in \mathbb{R}^d, \quad (1.1)$$

где каждая компонента  $m_i^{(j)}$  представляет значение  $j$ -й метрики в момент времени  $t_i$ .

Компоненты вектора наблюдений соответствуют различным категориям метрик, характеризующих работу отдельных подсистем видеоконвейера:

- **Метрики ML-конвейера:** *vidcap\_delay* (задержка видеозахвата), *vidcap\_fps* (частота кадров видеозахвата), *vidcap\_fps\_avg* (средняя частота кадров видеозахвата), характеризующие производительность модулей ком-

пьютерного зрения;

- **Метрики бэкенда:** *ml\_to\_backend\_kafka\_delay* (задержка передачи результатов ML через Kafka), *db\_insert\_delay* (время записи в базу данных), отражающие эффективность серверной части системы;
- **Метрики WebSocket-клиента:** *common\_event\_delay* (целевая end-to-end-задержка), *heartbeat\_\** (метрики жизнеспособности соединений), *event\_counter* (счетчики событий), *seq\_events\_health* (показатели корректности последовательности событий), характеризующие качество доставки результатов до конечных пользователей.

Для учета временных зависимостей в данных введем понятие скользящего окна наблюдений. Определим окно длины  $L$  и шаг сдвига  $s$ , где  $L$  представляет глубину истории, необходимую для прогнозирования, а  $s$  — частоту обновления прогнозов. Каждое  $k$ -е скользящее окно определяется как матрица:

$$X_k = [\mathbf{x}_{t_k-L+1}, \dots, \mathbf{x}_{t_k}] \in \mathbb{R}^{L \times d}, \quad (1.2)$$

содержащая  $L$  последовательных векторов наблюдений, предшествующих моменту прогнозирования.

Целевая переменная для задачи прогнозирования определяется как значение критической метрики end-to-end-задержки в будущий момент времени:

$$y_k = \text{common\_event\_delay}(t_k + \Delta), \quad (1.3)$$

где  $\Delta = 15$  с представляет горизонт прогнозирования, выбранный исходя из требований к заблаговременности предупреждений о потенциальных проблемах в системе.

Обучающая выборка для построения прогностической модели формируется как множество пар «окно-целевое значение»:

$$\mathcal{N} = \{(X_k, y_k)\}_{k=1}^N, \quad (1.4)$$

где  $N$  — общее количество доступных обучающих примеров, определяемое длиной исторических данных и параметрами скользящего окна.

В рамках данной постановки предполагается существование неизвестной целевой функции:

$$f^* : \mathbb{R}^{L \times d} \rightarrow \mathbb{R}, \quad (1.5)$$

которая отображает текущее состояние системы (представленное матрицей метрик скользящего окна) в прогнозируемое значение end-to-end-задержки.

Основная задача исследования состоит в построении алгоритма  $A : \mathbb{R}^{L \times d} \rightarrow \mathbb{R}$ , аппроксимирующего неизвестную функцию  $f^*$  с заданной точностью:

$$|A(X_k) - f^*(X_k)| \leq \varepsilon \quad \forall k, \quad (1.6)$$

где  $\varepsilon$  — допустимая погрешность прогнозирования, определяемая практическими требованиями к системе предупреждения.

К разрабатываемому алгоритму  $A$  предъявляется ряд критических требований, обусловленных спецификой применения в производственной среде:

1. **Точность прогнозирования:** минимизация среднеквадратической ошибки MSE на независимой валидационной выборке при обеспечении статистически значимого превосходства над базовыми методами прогнозирования;

2. **Производительность inference:** обеспечение времени отклика  $latency\_inference(A) < 1$  с при развертывании в стандартном Docker-контейнере с ограниченными вычислительными ресурсами;
3. **Адаптивность:** поддержка механизмов периодического дообучения на новых данных (warm-start инициализация, заморозка слоев, применение адаптеров/LoRA) для поддержания актуальности модели при изменении характеристик системы;
4. **Операционная гибкость:** возможность настройки частоты генерации прогнозов в диапазоне от онлайн-режима до фиксированных интервалов (например, каждые 30 минут) в зависимости от текущих требований к системе мониторинга.

Исходные данные для обучения и валидации алгоритма представляют собой многомерный временной ряд  $X \in \mathbb{R}^{n \times d}$  с элементами типа FLOAT64, формируемый из системы мониторинга Prometheus с периодичностью сбора 30 секунд. Объем доступных исторических данных составляет приблизительно 90643 точки, накопленные за период 16 дней непрерывной работы системы.

Итоговая формализация задачи: построить алгоритм  $A$ , наилучшим образом аппроксимирующий неизвестную функцию  $f^*$  и одновременно удовлетворяющий всем указанным ограничениям по точности, производительности и адаптивности для обеспечения надежного предиктивного мониторинга критических систем видеоаналитики.

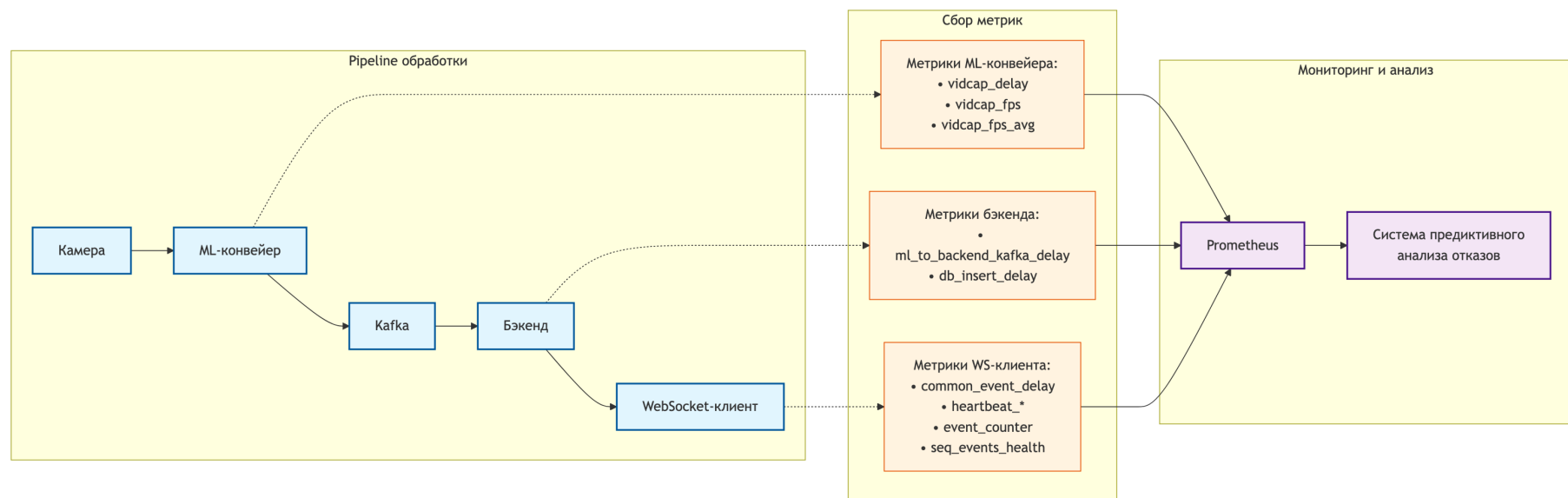


Рисунок 1.1 — Схема видеоконвейера и точки сбора метрик

## **2 Глава n**

### **2.1 Секция n**

### **3 Глава n**

#### **3.1 Секция n**



## **4 Глава n**

### **4.1 Секция n**

## ЗАКЛЮЧЕНИЕ

## СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

- 1) First
- 2) Second
- 3) Third

## Список сокращений и условных обозначений

### Сокращения:

API	Application Programming Interface — программный интерфейс приложения
Docker	платформа контейнеризации приложений
FPS	Frames Per Second — кадры в секунду
Kafka	Apache Kafka — распределенный брокер сообщений
LoRA	Low-Rank Adaptation — адаптация с низкоранговой аппроксимацией
ML	Machine Learning — машинное обучение
MLOps	Machine Learning Operations — операции машинного обучения
MSE	Mean Squared Error — среднеквадратическая ошибка
Prometheus	система мониторинга и оповещений с открытым исходным кодом
SLA	Service Level Agreement — соглашение об уровне обслуживания
WS	WebSocket — протокол полнодуплексной связи

### Условные обозначения:

$T$	множество временных меток наблюдений
$d$	число метрик, собираемых системой мониторинга
$L$	длина скользящего окна наблюдений
$s$	шаг сдвига скользящего окна
$\mathbf{x}_i$	$d$ -мерный вектор наблюдений в момент времени $t_i$
$X_k$	матрица скользящего окна размерности $L \times d$
$y_k$	целевая переменная (значение <i>common_event_delay</i> )
$\mathcal{N}$	обучающая выборка
$N$	общее количество обучающих примеров
$f^*$	неизвестная целевая функция
$A$	разрабатываемый алгоритм прогнозирования
$\varepsilon$	допустимая погрешность прогнозирования
$\Delta$	горизонт прогнозирования (15 секунд)
end-to-end	сквозной (от начала до конца процесса)
inference	процесс получения предсказаний от обученной модели
warm-start	инициализация обучения с предобученными параметрами