Научное издательство
Профессиональный Вестник

Информационные технологии и безопасность

Выпуск №2/2024

Научный журнал для лучших специалистов отрасли. Внутри: оригинальные работы по ИИ, анализу данных, облачным технологиям и инновациям в сфере ИТ.

ИНДЕКСАЦИИ ЖУРНАЛА

Google Scholar · International Scientific Indexing · Academic Resource Index ResearchBib · Научная электронная библиотека eLIBRARY.RU · CyberLeninka · CiteFactor Academic Scientific Journals

ISSN 3100-444X

support@professionalbulletinpublisher.com

professionalbulletinpublisher.com/

Brasov, Sat Sanpetru, Comuna Sanpetru, Str. Sfintii Constantin si Elena, nr. 6

Scientific publishing house
Professional Bulletin

Information Technology and Security

Issue №2/2024

A scientific journal for the best specialists in the industry. Inside: original works on AI, data analysis, cloud technologies and IT innovations.

INDEXATIONS

Google Scholar · International Scientific Indexing · Academic Resource Index ResearchBib · Научная электронная библиотека eLIBRARY.RU · CyberLeninka · CiteFactor Academic Scientific Journals

ISSN 3100-444X

support@professionalbulletinpublisher.com

professionalbulletinpublisher.com/

Brasov, Sat Sanpetru, Comuna Sanpetru, Str. Sfintii Constantin si Elena, nr. 6

Научное издательство «Профессиональный вестник»

Журнал «Профессиональный вестник. Информационные технологии и безопасность»

Профессиональный вестник. Информационные технологии и безопасность – профессиональное научное издание. Публикация в нем рекомендована практикам и исследователям, которые стремятся найти решения для реальных задач и поделиться своим опытом с профессиональным сообществом. Публикация в журнале подходит для тех специалистов, кто работает и активно развивает передовые ИТ-решения, такие как технологии ИИ, блокчейна, больших данных и другие.

Журнал рецензирует все входящие материалы. Рецензирование – двойное слепое, осуществляется внутренними и внешними рецензентами издательства. Статьи индексируются во множестве международных научных баз, доступ к базе данных журнала открыт для любого читателя. Публикация журнала происходит 4 раза в год.

Сайт издательства: https://www.professionalbulletinpublisher.com/

Выпуск № 2/2024

Жудец Брашов, Румыния

The scientific publishing house «Professional Bulletin»
Journal «Professional Bulletin. Information Technology and Security»

Professional Bulletin. Information Technology and Security is a professional scientific journal. The publication in it is recommended to practitioners and researchers who seek to find solutions to real-world problems and share their experiences with the professional community. The publication in journal is suitable for those specialists who work and actively develop advanced IT solutions, such as AI, blockchain, big data technologies and others.

The journal reviews all incoming materials. The review is double-blind, carried out by internal and external reviewers of the publishing house. Articles are indexed in a variety of international scientific databases, and access to the journal's database is open to any reader. Publication in the journal takes place 4 times a year.

Publisher's website: https://www.professionalbulletinpublisher.com/

Issue №2/2024
Brasov County, Romania

Содержание выпуска

Contents

UDC 004.8

A REVIEW OF REINFORCEMENT LEARNING ALGORITHMS FOR REAL-TIME PROBLEM SOLVING

Ismailov P.
master's degree, Al-Farabi Kazakh National University
(Almaty, Kazakhstan)

ОБЗОР АЛГОРИТМОВ ОБУЧЕНИЯ С ПОДКРЕПЛЕНИЕМ ДЛЯ РЕШЕНИЯ ПРОБЛЕМ В РЕАЛЬНОМ ВРЕМЕНИ

Исмаилов П.Н.
магистр, Казахский национальный университет имени аль-Фараби (Алматы, Казахстан)

Abstract

This article provides an overview of modern reinforcement learning (RL) algorithms used for solving real-time tasks. Various methods, including Q-learning, gradient algorithms, recurrent neural networks, and distributed learning, are analyzed, highlighting their capacity to adapt to changing environments and make effective decisions. Special attention is paid to computational resource optimization and model stability, which are critical for tasks requiring rapid response. Knowledge adaptation and transfer methods, such as multi-task learning, are also discussed as approaches that accelerate model training in data-scarce conditions. The application of these methods makes RL an effective tool for dynamic, high-tech fields, including autonomous control and robotics. The findings demonstrate the potential of RL in real-time conditions but also underline the need for further research to enhance algorithm resilience and reduce computational costs.

Keywords: reinforcement learning, adaptation, real-time algorithms, autonomous control, stability.

Аннотация

В статье представлен обзор современных алгоритмов обучения с подкреплением (ОП), применяемых для решения задач в режиме реального времени. Рассмотрены различные методы, включая Q-обучение, градиентные алгоритмы, рекуррентные нейронные сети и распределенное обучение, которые позволяют моделям адаптироваться к изменениям в окружающей среде и принимать эффективные решения. Особое внимание уделено проблемам оптимизации вычислительных ресурсов и обеспечению устойчивости моделей, что критично для задач с быстрым откликом. Также обсуждаются методы адаптации и переноса знаний, такие как мультизадачное обучение, которые позволяют ускорить обучение моделей в условиях недостатка данных. Результаты исследования демонстрируют потенциал использования ОП в условиях реального времени, однако также выявляют необходимость дальнейших исследований для решения задач, связанных с устойчивостью алгоритмов и снижением вычислительных затрат.

Ключевые слова: обучение с подкреплением, адаптация, алгоритмы реального времени, автономное управление, устойчивость.

Introduction

The modern development of artificial intelligence (AI) technologies has led to the need for adaptive algorithms capable of making real-time decisions. Reinforcement learning (RL), as one of the key AI directions, enables the creation of models that learn to interact with dynamic environments

and optimize actions based on accumulated experience. Unlike traditional learning methods, RL relies on reward and punishment principles, allowing models to independently develop strategies to achieve specified goals. This article aims to systematize RL methods for solving real-time problems and evaluate their effectiveness.

The main tasks that RL addresses include building models capable of considering environmental uncertainties and changing conditions, which is especially relevant for areas such as unmanned vehicle control, robotics, and automated decision-making systems. RL requires handling large volumes of data and adapting models to real-time environmental changes. This article explores algorithms focused on high performance and minimal response time, allowing them to be successfully applied in various practical fields.

Introducing RL methods in real-time tasks requires hybrid approaches combining classical learning algorithms and the latest deep learning technologies. This article examines various approaches to the development and application of RL in real-time, analyzing their strengths and limitations. The study aims to provide an overview of modern RL algorithms and identify the most effective solutions for applications that require rapid system response.

Main part

RL algorithms are divided into several categories, with the most interest in Q-learning and gradient-based methods. Q-learning, one of the basic RL algorithms, enables efficient agent training using state tables; however, as environment complexity grows, this method encounters the "curse of dimensionality" problem [1]. For real-time tasks, algorithms that can quickly adapt to environmental changes are crucial. An example is gradient-based methods, which optimize agent behavior by minimizing loss functions. Algorithms like Policy Gradient and Actor-Critic focus on continuous strategy updates, making them suitable for high-dynamic tasks. Figure 1 shows the architecture of the Actor-Critic model, widely used in real-time tasks.

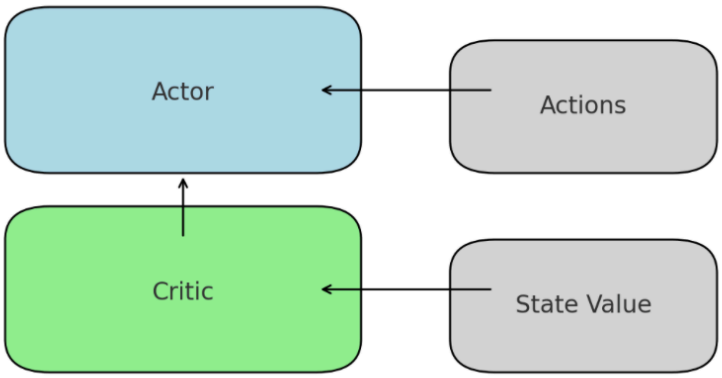

Figure 1. Actor-Critic model architecture

This figure shows the interaction between the critic, evaluating the current state, and the actor, adjusting the agent's actions.

Another promising area involves algorithms that use recurrent neural networks (RNNs) capable of capturing temporal dependencies in data [2]. Using RNNs in RL allows models to remember previous states, which is particularly useful for sequence-based tasks. An example implementation is the Recurrent Q-Learning algorithm, which effectively trains agents in conditions where state information changes over time. A critical aspect of applying RL to real-time tasks is optimizing computational resources. Since tasks requiring rapid response have high performance requirements, methods that implement parallel computing, such as Distributed Reinforcement Learning, are popular. In such systems, computations are distributed across multiple processors or machines, significantly increasing training speed [3].

An example of real-time RL usage is in unmanned vehicle control, where the algorithm must quickly adapt to road changes and make decisions in unpredictable environments. Algorithms combining RL and deep learning, such as Deep Deterministic Policy Gradient (DDPG), are used for

these tasks. DDPG optimizes agent behavior for continuous action tasks, making it effective in dynamic environments. To increase stability and improve training outcomes, algorithms with prioritization mechanisms, such as Prioritized Experience Replay, are applied. In this method, significant events carry more weight in the learning process, enabling the algorithm to adapt more quickly to environmental changes. This approach is used in systems with high stability and reliability requirements [4, 5].

Adaptation and knowledge transfer in real-time tasks

In real-time RL applications, a critical task remains adapting algorithms to new conditions and their ability to transfer knowledge between different tasks. During model development and training, the need often arises to accelerate the learning process by using knowledge already obtained in similar conditions [6-8]. This approach significantly reduces time and computational costs, allowing the model to quickly reach the required level of adaptability.

Knowledge transfer is implemented through methods that include retraining on new data using previous models or combining experience from multiple sources. A solution to this problem is multitask learning, where the model simultaneously trains on several tasks, with the ability to identify and use common patterns. This approach is especially useful when tasks have similar elements or goals, allowing the model to "switch" between tasks while retaining the achieved level of competence [9].

The implementation of knowledge transfer methods requires not only structural adaptation of the model but also controlling the possibility of excessive accumulation of "old" knowledge, which may lose relevance. A crucial aspect is a mechanism for tracking the significance of previously obtained data and updating it according to new task requirements. This dynamic data control system enables models to handle changing conditions effectively and avoid situations where accumulated knowledge begins to reduce performance in new tasks [10].

In real-time tasks, knowledge transfer plays an important role in cases of data scarcity and situations where training "from scratch" is difficult due to time or resource constraints. Knowledge transfer enables models to adapt more quickly and function effectively in conditions close to real, making it an integral part of RL applications in dynamic environment tasks.

Conclusion

Reinforcement learning algorithms offer extensive possibilities for creating adaptive models capable of solving tasks in real-time. The methods discussed, including Q-learning, gradient approaches, the use of recurrent neural networks, and distributed learning, demonstrate high efficiency when interacting with dynamic environments. These algorithms can quickly adapt to environmental changes, making them especially useful for areas like autonomous control and robotics.

When using RL methods, it is essential to consider complexities related to optimizing computational resources and ensuring model stability. Adapting algorithms to real-world conditions requires prioritization methods and hybrid approaches involving deep learning. Knowledge transfer also promotes efficient algorithm implementation in situations where comprehensive training "from scratch" is impossible. Transfer mechanisms and multitask learning significantly reduce training costs and accelerate model adaptation.

In general, the application of RL methods in real-time tasks opens prospects for their use in high-tech and dynamic fields. However, further research tasks remain, such as developing more robust algorithms, reducing computational costs, and improving quality control methods. Continuing progress in this field will enable integrating RL into more complex and variable conditions, expanding the practical applications of these methods.

References

1. Kulida E.L., Lebedev V.G. Methods for solving planning and air traffic flow control tasks. Part 2. Application of deep reinforcement learning methods // Control Problems. 2023. Vol. 2. P. 3-18.

2. Orlova E.V. Reinforcement learning as an artificial intelligence technology for solving socio-economic problems: evaluation of algorithm performance // π-Economy. 2023. Vol. 16. No.5. P. 38-50.
3. Eremeev A.P., Kozhukhov A.A. Implementation of reinforcement learning methods based on temporal differences and a multi-agent approach for real-time intelligent systems // Software Products and Systems. 2017. Vol. 30. No.1. P. 28-33.
4. Andronov S.A., Prokofeva M.S. Comparative evaluation of the effectiveness of reinforcement learning and optimization algorithms in transport process management in the Anylogic environment // Systems Analysis and Logistics: Journal. 2022. No.2. P. 32.
5. Agarkov Y.Yu. Machine learning methods for optimizing the design of neuromorphic systems // Innovations and Investments. 2023. No.6. P. 313-319.
6. Zaitseva Yu.S. Artificial intelligence methods for solving control tasks in robotic and mechatronic systems: a review // Proceedings of Higher Educational Institutions. Mechanical Engineering. 2024. No.1(766). P. 41-56.
7. Mohammed H.A.R., Zargaryan E.V., Zargaryan Yu.A. Machine learning and deep learning models for electronic information security in mobile networks // Proceedings of the Southern Federal University. Technical Sciences. 2022. No.3(227). P. 211-222.
8. Powell K.M., Machalek D., Quah T. Real-time optimization using reinforcement learning // Computers & Chemical Engineering. 2020. Vol. 143. P. 107077.
9. Shyalika C., Silva T., Karunananda A. Reinforcement learning in dynamic task scheduling: A review // SN Computer Science. 2020. Vol. 1. No.6. P. 306.
10. Padakandla S. A survey of reinforcement learning algorithms for dynamically varying environments // ACM Computing Surveys (CSUR). 2021. Vol. 54. No.6. P. 1-25.

УДК 004.8

ГЕНЕРАЦИЯ СИНТЕТИЧЕСКИХ ДАННЫХ В ОБУЧЕНИИ ИСКУССТВЕННЫХ НЕЙРОННЫХ СЕТЕЙ

Головин А.С.
магистр, Южный федеральный университет
(Ростов-на-Дону, Россия)

SYNTHETIC DATA GENERATION IN TRAINING ARTIFICIAL NEURAL NETWORKS

Golovin A.
master's degree, Southern Federal University
(Rostov-on-Don, Russia)

Аннотация

В статье представлен обзор методов генерации синтетических данных для обучения искусственных нейронных сетей (ИНС) в условиях ограниченного доступа к реальным данным. Рассмотрены ключевые подходы, включая генеративные состязательные сети (GAN), методы повышения данных и статистические модели, применимые к различным типам данных. Особое внимание уделено применению синтетических данных в задачах распознавания лиц, диагностики редких заболеваний и управления автономными транспортными системами. Проанализированы преимущества и ограничения каждого метода, а также их влияние на качество и точность моделей. Рассмотрены возможные риски, связанные с использованием синтетических данных, такие как искажения и смещения, и подходы к их минимизации с целью повышения надежности обучения. Применение синтетических данных открывает значительные перспективы для расширения возможностей ИНС, способствуя улучшению их эффективности и обобщающей способности в реальных задачах.

Ключевые слова: синтетические данные, обучение с подкреплением, генеративные состязательные сети, искусственные нейронные сети, повышение данных.

Abstract

This article presents an overview of synthetic data generation methods for training artificial neural networks (ANNs) under limited access to real data. Key approaches such as Generative Adversarial Networks (GAN), data augmentation methods, and statistical models applicable to various data types are reviewed. Special attention is given to the application of synthetic data in face recognition, rare disease diagnosis, and autonomous systems management. The advantages and limitations of each method, as well as their impact on model accuracy, are analyzed. Potential risks associated with synthetic data, including biases and distortions, and approaches to mitigate these issues to enhance model reliability, are also discussed. The use of synthetic data provides substantial opportunities for advancing ANNs, improving their effectiveness and generalizability in practical tasks.

Keywords: synthetic data, reinforcement learning, generative adversarial networks, artificial neural network, data augmentation.

Введение

С развитием технологий машинного обучения (МО) и искусственных нейронных сетей (ИНС) возникла потребность в большом количестве данных для обучения моделей. Однако

реальные данные не всегда доступны в требуемом объеме, что усложняет процесс обучения и снижает точность прогнозов моделей. В таких условиях генерация синтетических данных становится важным инструментом, позволяющим восполнить нехватку данных и повысить эффективность ИНС. Основной целью данной статьи является исследование методов генерации синтетических данных, используемых для обучения ИНС, и оценка их эффективности в различных задачах.

Использование синтетических данных позволяет не только восполнить дефицит информации, но и улучшить производительность ИНС за счет формирования большего разнообразия обучающих примеров. Синтетические данные могут быть сгенерированы различными способами, включая статистические модели, генеративные состязательные сети (ГСН) и методы на основе преобразования изображений. Генерация синтетических данных имеет значительный потенциал для применения в областях, где получение реальных данных затруднено, например, в медицине, финансовом секторе и анализе изображений. В данной статье рассматриваются основные методы генерации синтетических данных и их влияние на обучение ИНС.

Введение синтетических данных требует также учета рисков, связанных с их использованием, таких как возможность появления смещений в данных и снижение обобщающей способности модели. Одной из задач данной статьи является изучение потенциальных рисков, связанных с применением синтетических данных, а также рассмотрение подходов к их минимизации. Целью исследования является обобщение текущих подходов к генерации синтетических данных и выявление наиболее эффективных методов для конкретных задач ИНС, а также анализ проблем, возникающих при внедрении синтетических данных в процесс обучения.

Основная часть

Методы генерации синтетических данных могут быть разделены на несколько подходов в зависимости от типа данных и целей их использования. Одним из наиболее распространенных методов является применение генеративных состязательных сетей (Generative Adversarial Networks, GAN), которые состоят из двух нейронных сетей: генератора и дискриминатора. Генератор создает синтетические данные, стараясь сделать их максимально похожими на реальные, в то время как дискриминатор пытается отличить синтетические данные от реальных [1]. Совместное обучение этих сетей позволяет генератору постепенно улучшать качество синтетических данных. Метод ГСН широко используется для генерации изображений, текстов и других сложных данных.

Для задач, связанных с изображениями, также применяют методы повышения данных (Data Augmentation), такие как поворот, масштабирование и добавление шума к исходным изображениям [2]. Эти методы позволяют значительно расширить обучающую выборку, не требуя создания новых данных, что особенно полезно для задач классификации и распознавания. В таблице 1 представлено сравнение различных методов генерации синтетических данных, где указаны области применения, преимущества, ограничения и примеры использования.

Таблица 1 [3]

Основные методы генерации синтетических данных, их применение, преимущества и ограничения

Метод	**Применение**	**Преимущества**	**Ограничения**	**Примеры применения**
ГСН	Изображения, текст, звуковые данные	Высокое качество, способность к обучению сложных распределений	Длительное обучение, необходимость тонкой настройки	Генерация лиц для распознавания, создание текстовых описаний

Повышение данных	Изображения	Увеличение выборки, простота, низкая стоимость	Ограниченная генерация новых признаков	Обучение классификаторов изображений, медицинская визуализация
Статистические модели	Табличные данные, финансовые данные	Контроль над распределениями, предсказуемость характеристик	Могут не учитывать сложные зависимости	Анализ финансовых данных, моделирование поведения пользователей
Модели на основе временных рядов (PHH, LSTM)	Временные ряды, данные сенсоров	Способность учитывать временные зависимости	Требует большого объема данных, высокая вычислительная нагрузка	Прогнозирование спроса, анализ данных IoT
Симуляции на основе физических моделей	Инженерия, физические процессы	Реализм, высокое соответствие физическим законам	Высокие затраты на разработку, сложность моделей	Обучение автономных систем, робототехника

В задачах, требующих использования табличных данных, часто применяются статистические модели, которые позволяют генерировать данные на основе заранее определенных вероятностных распределений. Этот метод обеспечивает высокую точность синтетических данных, соответствующих реальным параметрам, что особенно полезно при создании финансовых или пользовательских данных [4].

На рисунке 1 представлена сравнительная эффективность различных методов генерации синтетических данных, таких как ГСН, повышение данных, статистические модели, LSTM и физические симуляции. Эти данные помогают оценить преимущества каждого подхода.

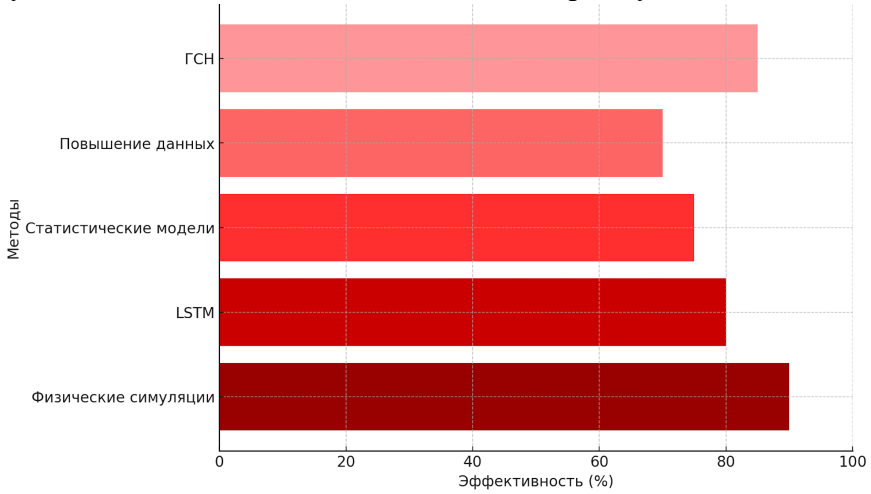

Рисунок 1. Эффективность методов генерации синтетических данных

Как видно из рисунка 1, физические симуляции показывают наибольшую эффективность (90%), благодаря их способности учитывать реальные физические процессы. Методы ГСН (85%) и LSTM (80%) также демонстрируют высокую результативность, особенно в задачах, требующих сложного моделирования данных. Повышение данных и статистические модели, несмотря на свою простоту и доступность, имеют эффективность 70% и 75% соответственно, что делает их удобными для быстрого увеличения обучающих выборок. Эти показатели подчеркивают необходимость выбора подходящего метода в зависимости от задачи и ресурсов.

Пример кода для генерации табличных синтетических данных с нормальным распределением представлен ниже.

```
import numpy as np
import pandas as pd

# Генерация синтетических данных на основе нормального распределения
mean = [0, 0]
cov = [[1, 0.5], [0.5, 1]]
synthetic_data = np.random.multivariate_normal(mean, cov, size=100)
df = pd.DataFrame(synthetic_data, columns=["Feature1", "Feature2"])

print(df.head())
```

Для задач временных рядов используются рекуррентные нейронные сети и модели с долгой краткосрочной памятью (LSTM), что позволяет учитывать временные зависимости в данных. Этот метод особенно полезен для прогнозирования и моделирования процессов, зависящих от последовательности событий, таких как изменения спроса или температурные колебания. Важно отметить, что использование таких методов требует значительных вычислительных ресурсов и больших объемов данных для обучения.

Использование синтетических данных в процессе обучения ИНС требует учета ряда рисков. Например, при генерации синтетических данных возможно появление искажений, что может привести к снижению точности и надежности модели [5]. Для минимизации этих рисков часто применяются гибридные подходы, совмещающие реальные и синтетические данные. Такой метод обучения позволяет повысить обобщающую способность модели, избегая излишней зависимости от синтетических данных.

Применение синтетических данных предоставляет значительные возможности для обучения ИНС в условиях ограниченного доступа к реальным данным. Так, в медицине они могут быть использованы для моделирования редких заболеваний, что позволяет создавать более точные диагностические модели и системы прогнозирования.

Практическое применение синтетических данных в обучении ИНС

Для задач, связанных с распознаванием лиц и анализа изображений, синтетические данные, созданные с помощью GAN, позволяют обучать модели на разнообразных примерах, сохраняя при этом высокое качество генерации. Например, в задачах безопасности GAN могут генерировать синтетические изображения лиц для тренировок систем распознавания в условиях ограниченных данных.

В медицине синтетические данные находят применение при обучении моделей для диагностики редких заболеваний [6]. При недостаточности реальных данных модели могут использовать синтетические изображения, созданные на основе исходных снимков, чтобы расширить обучающую выборку и повысить точность диагностики. В частности, синтетические рентгеновские снимки или МРТ-изображения позволяют обучить нейронные сети без необходимости получения новых данных от пациентов, что снижает риски конфиденциальности и затраты.

В автономных транспортных системах генерация данных также играет ключевую роль. Для беспилотных автомобилей синтетические изображения дорожных ситуаций помогают моделям обучаться на различных сценариях, включая экстремальные погодные условия или неожиданные препятствия. Использование GAN для создания таких данных помогает избежать необходимости проведения длительных и дорогостоящих полевых испытаний.

В финансовом секторе ИНС, обученные на синтетических данных, используются для анализа поведения пользователей, прогнозирования рыночных трендов и оценки рисков. Статистические модели позволяют генерировать данные с учетом заданных распределений, что полезно при моделировании редких событий, таких как кризисы или внезапные рыночные колебания [7]. Использование синтетических данных в процессе обучения ИНС предоставляет значительные возможности для обучения моделей в условиях ограниченного доступа к реальным данным.

Реализация генерации синтетических данных с помощью GAN

GAN играют важную роль в создании синтетических данных, особенно в задачах, связанных с генерацией изображений, текста и других сложных данных. GAN представляют собой архитектуру, состоящую из двух нейронных сетей — генератора и дискриминатора. Генератор создает синтетические данные, используя случайный шум, а дискриминатор оценивает, насколько близки созданные данные к реальным. Процесс обучения GAN заключается в том, что обе сети "состязаются" друг с другом: генератор старается создавать данные, неотличимые от реальных, а дискриминатор стремится эффективно различать синтетические и реальные данные. В результате генератор обучается создавать все более качественные данные, что делает этот метод особенно эффективным для задач, требующих реалистичной генерации [8, 9].

В следующем примере показан процесс генерации синтетических изображений с использованием GAN. На вход генератору подается случайный шум, на основе которого он создает изображение. Дискриминатор, в свою очередь, оценивает это изображение, указывая, насколько оно похоже на реальное. После завершения обучения генератор может использоваться для создания синтетических изображений, которые могут быть полезны для обучения других моделей, например, в задачах распознавания объектов или классификации изображений.

```
import tensorflow as tf
from tensorflow.keras.layers import Dense, Reshape, Flatten, LeakyReLU
from tensorflow.keras.models import Sequential

# Параметры
latent_dim = 100  # Размерность скрытого пространства (зашумленных данных)

# Модель генератора
def build_generator():
    model = Sequential()
    model.add(Dense(256, input_dim=latent_dim))
    model.add(LeakyReLU(alpha=0.2))
    model.add(Dense(512))
    model.add(LeakyReLU(alpha=0.2))
    model.add(Dense(1024))
    model.add(LeakyReLU(alpha=0.2))
    model.add(Dense(28 * 28, activation='tanh'))
    model.add(Reshape((28, 28, 1)))
    return model

# Модель дискриминатора
def build_discriminator():
    model = Sequential()
    model.add(Flatten(input_shape=(28, 28, 1)))
    model.add(Dense(512))
    model.add(LeakyReLU(alpha=0.2))
    model.add(Dense(256))
    model.add(LeakyReLU(alpha=0.2))
    model.add(Dense(1, activation='sigmoid'))
    return model

# Компиляция моделей
generator = build_generator()
discriminator = build_discriminator()
discriminator.compile(optimizer='adam', loss='binary_crossentropy', metrics=['accuracy'])
```

```
# GAN - совмещенная модель
discriminator.trainable = False  # Фиксируем дискриминатор для обучения только генератора
gan_input = tf.keras.Input(shape=(latent_dim,))
generated_image = generator(gan_input)
gan_output = discriminator(generated_image)
gan = tf.keras.Model(gan_input, gan_output)
gan.compile(optimizer='adam', loss='binary_crossentropy')

# Пример тренировки GAN
import numpy as np

# Случайный шум для генерации синтетических данных
noise = np.random.normal(0, 1, (1, latent_dim))
generated_image = generator.predict(noise)

# Визуализация сгенерированного изображения
import matplotlib.pyplot as plt

plt.imshow(generated_image[0, :, :, 0], cmap='gray')
plt.title("Сгенерированное изображение")
plt.axis('off')
plt.show()
```

Этот пример демонстрирует, как использовать GAN для создания синтетических изображений на основе случайного шума. На этапе визуализации получаем изображение, сгенерированное генератором, что наглядно показывает, как нейронная сеть способна воссоздать структурированные данные на основе случайных входных данных. Данная архитектура широко применяется в задачах, где синтетические данные необходимы для обучения моделей, работающих в условиях дефицита реальных данных, включая задачи, связанные с медицинской визуализацией, автономными транспортными системами и биометрией.

Заключение

Использование синтетических данных, созданных с помощью генеративных алгоритмов, значительно расширяет возможности обучения искусственных нейронных сетей, особенно в условиях ограниченного доступа к реальным данным. Такие методы, как GAN, позволяют создавать качественные и реалистичные данные, пригодные для обучения моделей в задачах распознавания лиц, диагностики заболеваний, управления автономными системами и финансового анализа. Объединение различных подходов, включая методы повышения данных и статистические модели, делает возможным решение широкого спектра задач с высокой точностью.

Особое внимание необходимо уделять возможным искажениям и смещениям, возникающим при генерации синтетических данных, так как они могут негативно повлиять на точность и обобщающую способность моделей. Применение гибридных подходов, совмещающих реальные и синтетические данные, позволяет снизить такие риски и обеспечить более надежное обучение. Важно также учитывать, что генерация синтетических данных требует значительных вычислительных ресурсов, особенно при использовании рекуррентных сетей и физических симуляций.

Синтетические данные становятся все более востребованными в современных областях науки и технологий, и их применение открывает перспективы для разработки новых решений и алгоритмов. При дальнейшем совершенствовании методов генерации и контроля качества синтетических данных можно ожидать существенного увеличения точности и эффективности

ИНС, что будет способствовать развитию МО и его внедрению в различные прикладные задачи.

Список литературы

1. Пчелинцев С., Юляшков М.А., Ковалева О.А. Метод создания синтетических наборов данных для обучения нейросетевых моделей распознаванию объектов // Информационно-управляющие системы. 2022. №3(118). С. 9-19.
2. Ходасевич Л.А. Генерация реалистичных изображений для обучения искусственных нейронных сетей в задаче навигации робота // Информатика. 2018. Т. 15. №4. С. 50-58.
3. Моисеев Б., Чигорин А. Классификация автодорожных знаков на основе свёрточной нейросети, обученной на синтетических данных // The 22nd International Conference on Computer Graphics and Vision. 2012. С. 284-287.
4. Рубцов И.А. Методические подходы к исправлению проблемы недостаточности инфографических данных для обучения нейронных сетей // Вестник магистратуры. 2020. №5-3. С. 108.
5. Ковалев В.А., Козловский С.А., Калиновский А.А. Генерация искусственных рентгеновских изображений грудной клетки с использованием генеративно-состязательных нейронных сетей // Информатика. 2018. Т. 15. №2. С. 7-16.
6. Кабанова В.В., Логунова О.С. Применение искусственного интеллекта при работе с мультимедийной информацией // Вестник Череповецкого государственного университета. 2022. №6(111). С. 23-41.
7. Малов Д.А., Летенков М.А. Методика генерации искусственных наборов данных и архитектура системы распознавания лиц для взаимодействия с роботами внутри киберфизического пространства // Робототехника и техническая кибернетика. 2019. Т. 7. №2. С. 100-108.
8. Берзин В.И., Судейкин М.И. Разработка алгоритмов генерации синтетических данных для обучения нейросетевых моделей детектирования объектов на изображении // Физико-техническая информатика (СРТ2020). 2020. С. 106-122.
9. Юрин А.Н. Создание обучающей выборки для искусственной нейронной сети системы технического зрения // Механизация и электрификация сельского хозяйства. 2023. Т. 1. №56. С. 148-153.

УДК 004.9

СРАВНИТЕЛЬНЫЙ АНАЛИЗ БЛОКЧЕЙН-ПЛАТФОРМ ДЛЯ ФИНАНСОВЫХ ТРАНЗАКЦИЙ

Шелест Н.В.
Новосибирский государственный университет
(Новосибирск, Россия)

COMPARATIVE ANALYSIS OF BLOCKCHAIN PLATFORMS FOR FINANCIAL TRANSACTIONS

Shelest N.
Novosibirsk State University
(Novosibirsk, Russia)

Аннотация

В данной статье представлен сравнительный анализ блокчейн-платформ, используемых для финансовых транзакций, с целью выявления их ключевых характеристик и областей применения. Рассмотрены такие платформы, как Ethereum, Hyperledger Fabric, Ripple и Stellar, которые различаются по алгоритмам консенсуса, времени обработки транзакций, пропускной способности и уровню децентрализации. Обсуждаются аспекты безопасности и устойчивости, влияющие на эффективность блокчейн-решений в финансовом секторе, а также перспективы использования различных платформ для открытых и корпоративных систем. Особое внимание уделено анализу параметров, таких как масштабируемость, энергопотребление и интеграция с существующими финансовыми сервисами. Результаты исследования показывают, что выбор блокчейн-платформы зависит от специфики задач и требований к безопасности, скорости и уровню децентрализации. Применение блокчейн-технологий в финансовом секторе продолжает развиваться, что открывает возможности для повышения надежности и эффективности транзакций.

Ключевые слова: блокчейн, финансовые транзакции, алгоритм консенсуса, безопасность, децентрализация.

Abstract

This article presents a comparative analysis of blockchain platforms used for financial transactions, aimed at identifying their key characteristics and application areas. Platforms such as Ethereum, Hyperledger Fabric, Ripple, and Stellar are reviewed, with differences in consensus algorithms, transaction processing times, throughput, and decentralization levels highlighted. Security and resilience aspects affecting the effectiveness of blockchain solutions in the financial sector are discussed, along with prospects for using various platforms in both open and corporate systems. Special focus is placed on analyzing scalability, energy consumption, and integration with existing financial services. The study results indicate that the choice of blockchain platform depends on task specifics and requirements for security, speed, and decentralization level. Blockchain applications in the financial sector are evolving, providing opportunities to enhance the reliability and efficiency of transactions.

Keywords: blockchain, financial transactions, consensus algorithm, security, decentralization.

Введение

С распространением цифровых технологий и развитием финансовых транзакций возросла потребность в безопасных и эффективных методах передачи данных и выполнения

операций. Блокчейн-технологии, которые впервые обрели популярность благодаря криптовалютам, стали основой для создания защищенных децентрализованных систем, обеспечивающих прозрачность и устойчивость к вмешательству. В частности, для финансовых операций блокчейн открывает новые возможности по обеспечению прозрачности и защите данных от несанкционированных изменений. Целью данной статьи является проведение сравнительного анализа различных блокчейн-платформ, используемых для финансовых транзакций, с акцентом на их ключевые характеристики и области применения.

Одной из значительных проблем, с которой сталкиваются организации, является выбор подходящей блокчейн-платформы, соответствующей их потребностям и обеспечивающей требуемый уровень безопасности и производительности. В настоящее время существует множество блокчейн-платформ, каждая из которых имеет свои особенности в плане архитектуры, пропускной способности и алгоритмов консенсуса. В рамках данной работы будут рассмотрены наиболее популярные блокчейн-платформы, такие как Ethereum, Hyperledger Fabric и Ripple, применяемые в финансовом секторе. Их анализ позволит выявить преимущества и недостатки каждой платформы и сформировать рекомендации для их использования в зависимости от специфики задачи.

Для обеспечения объективности в сравнительном анализе будут проанализированы ключевые параметры, такие как скорость транзакций, безопасность, масштабируемость и уровень децентрализации. Эти показатели играют важную роль в оценке эффективности блокчейн-платформ для финансовых операций, так как определяют, насколько быстро и безопасно может быть обработана транзакция. Исследование также затронет вопросы перспектив развития блокчейн-технологий в финансовой сфере, учитывая возможные инновации и адаптацию платформ к новым требованиям.

Основная часть

Одним из ключевых аспектов выбора блокчейн-платформы для финансовых транзакций является алгоритм консенсуса, который напрямую влияет на скорость, безопасность и энергопотребление сети. Например, платформа Ethereum использует алгоритм Proof of Work (PoW), который хотя и обеспечивает высокий уровень безопасности, характеризуется низкой производительностью и значительным потреблением ресурсов. Переход на Proof of Stake (PoS), который планируется для Ethereum, должен улучшить эти показатели, повысив скорость транзакций и снизив энергозатраты [1]. Hyperledger Fabric, напротив, применяет алгоритм Practical Byzantine Fault Tolerance (PBFT), ориентированный на закрытые сети с меньшим числом доверенных узлов, что обеспечивает высокую скорость обработки транзакций. Ripple использует собственный алгоритм Ripple Protocol Consensus Algorithm (RPCA), оптимизированный для межбанковских транзакций и поддерживающий быструю обработку данных [2].

На рисунке 1 показано соотношение средней скорости транзакции и пропускной способности различных блокчейн-платформ, используемых в финансовых операциях. Диаграмма позволяет наглядно оценить, как ключевые параметры платформ влияют на их производительность и область применения.

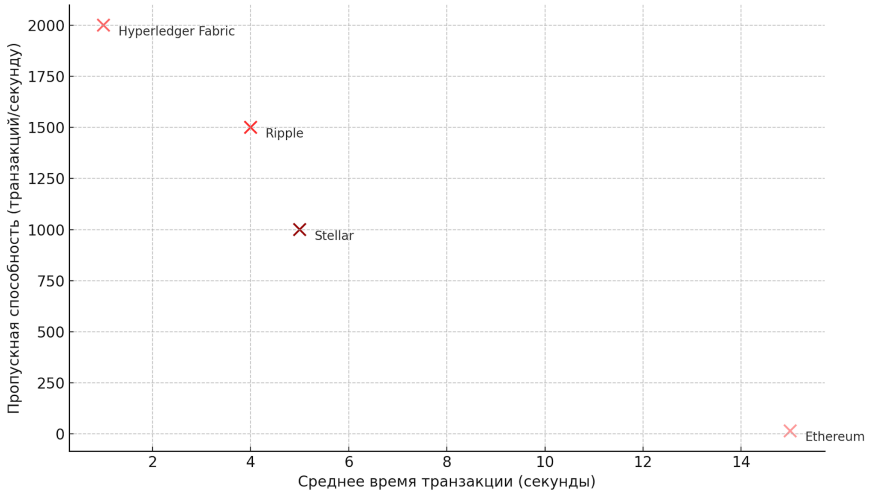

Рисунок 1. Характеристики блокчейн-платформ для финансовых операций

Как видно из рисунка 1, платформы значительно различаются по времени обработки транзакций и пропускной способности. Hyperledger Fabric демонстрирует наилучшую пропускную способность (2000 транзакций в секунду) при минимальном времени транзакции (<1 секунда), что делает его предпочтительным выбором для корпоративных систем. Ripple, с пропускной способностью 1500 транзакций в секунду и временем транзакции 4 секунды, ориентирован на межбанковские платежи. Stellar и Ethereum обеспечивают средние показатели, где Ethereum из-за своей высокой децентрализации имеет значительное время обработки транзакции (15 секунд), но остаётся популярным для децентрализованных приложений. Эти данные подчёркивают необходимость выбора платформы в зависимости от требований к скорости, масштабируемости и области использования.

Для сравнения в таблице 1 приведены характеристики трех популярных блокчейн-платформ, используемых в финансовых транзакциях. Эти платформы различаются по уровню безопасности, масштабируемости, скорости обработки транзакций и доступности инструментов для разработки, что позволяет выбрать оптимальное решение в зависимости от потребностей конкретного бизнеса.

Таблица 1

Сравнение характеристик блокчейн-платформ, используемых для финансовых транзакций

Платформа	**Алгоритм консенсуса**	**Среднее время транзакции**	**Пропускная способность (транзакций/сек)**	**Уровень децентрализации**	**Использование в финансах**
Ethereum	PoW / переход на PoS	15 сек	-15	Высокий	Криптовалюты, децентрализованные приложения
Hyperledger Fabric	PBFT	<1 сек	2000	Низкий	Корпоративные сети, приватные транзакции
Ripple	RPCA	4 сек	1500	Средний	Международные платежи, обмен валют
Stellar	SCP	5 сек	1000	Средний	Финансовые трансакции, микроплатежи

Помимо времени обработки транзакций и алгоритма консенсуса, важным фактором является уровень децентрализации. Ethereum характеризуется высоким уровнем

децентрализации благодаря большому количеству независимых узлов, однако это снижает его масштабируемость и увеличивает энергозатраты, что ограничивает его применение в корпоративных системах с высокой нагрузкой. Hyperledger Fabric, напротив, используется преимущественно в частных корпоративных сетях и поддерживает более централизованное управление, что позволяет достигать высокой производительности и низкого уровня энергопотребления, но ограничивает возможности его использования в публичных децентрализованных приложениях [3].

Пропускная способность платформ также играет важную роль для финансовых операций, где высокие объемы транзакций требуют быстрого отклика и стабильности сети. Например, Hyperledger Fabric и Ripple достигают высокой пропускной способности благодаря централизованным решениям и контролируемым узлам, что делает их подходящими для крупных корпоративных сетей и межбанковских расчетов. Stellar, использующий алгоритм SCP (Stellar Consensus Protocol), ориентирован на поддержку финансовых транзакций и микроплатежей, что делает его гибким и доступным для мелких финансовых операций.

Энергопотребление является еще одним важным параметром, особенно актуальным в контексте устойчивого развития. Платформы с алгоритмами PoW, такие как Ethereum, потребляют значительно больше энергии по сравнению с Hyperledger Fabric и Ripple, использующими менее энергозатратные алгоритмы. Для финансовых организаций, которые ставят целью минимизацию затрат и сокращение углеродного следа, это может стать критерием выбора блокчейн-платформы [4].

Каждая из этих платформ имеет уникальные особенности, подходящие для различных финансовых приложений. Ethereum, благодаря своей децентрализации и поддержке смарт-контрактов, остается популярной в секторе криптовалют и децентрализованных финансов. Hyperledger Fabric и Ripple, напротив, лучше подходят для корпоративных решений и межбанковских транзакций, где важна высокая скорость и контроль за сетью.

Анализ параметров безопасности и устойчивости блокчейн-платформ для финансовых транзакций

Безопасность и устойчивость являются критически важными характеристиками блокчейн-платформ, особенно для финансовых транзакций, где требуется защищать данные от несанкционированного доступа и внешних атак. Платформы, ориентированные на финансовые операции, должны обеспечивать надежные механизмы защиты, включающие криптографические протоколы, устойчивость к взломам, защиту от атак типа «двойное расходование» и эффективное управление узлами.

Ethereum, как одна из наиболее децентрализованных платформ, использует криптографические механизмы, которые обеспечивают высокий уровень безопасности. Благодаря своему алгоритму консенсуса PoW (с переходом на PoS), Ethereum демонстрирует устойчивость к внешним атакам, так как изменения в цепи блоков требуют значительных вычислительных ресурсов. При этом децентрализованная структура сети снижает вероятность атак типа Sybil (атака с множественными узлами) [5]. Однако высокая степень децентрализации также приводит к замедлению подтверждения транзакций и увеличению риска сбоев при высоких нагрузках.

Hyperledger Fabric, в отличие от Ethereum, ориентирован на приватные корпоративные сети, что снижает вероятность атаки за счет использования доверенных узлов и более узкого круга участников. Этот подход позволяет реализовать контролируемую среду с повышенным уровнем безопасности и гибкости. Например, Hyperledger Fabric предоставляет возможность управления правами доступа и использования каналов для ограниченного обмена информацией, что делает его эффективным инструментом для приватных финансовых операций [6]. Однако этот централизованный подход снижает уровень децентрализации, делая платформу менее устойчивой к внешним атакам в сравнении с публичными сетями.

Ripple, специально разработанный для межбанковских и международных транзакций, применяет алгоритм RPCA, который обеспечивает высокую скорость транзакций и устойчивость к сбоям. Ripple концентрируется на поддержании стабильности системы,

ориентируясь на надежные связи с банками и финансовыми учреждениями. Этот подход обеспечивает защиту данных на высоком уровне, хотя и уступает Ethereum по уровню децентрализации.

Stellar также имеет высокий уровень безопасности за счет применения собственного алгоритма консенсуса SCP (Stellar Consensus Protocol), который ориентирован на эффективное управление узлами и защиту от атак. Этот протокол обеспечивает быструю обработку микроплатежей, что делает его оптимальным для финансовых систем с небольшими транзакциями [7]. В отличие от Hyperledger Fabric, Stellar ориентирован на более широкий круг участников, сохраняя гибкость и доступность для различных организаций, что делает его оптимальным для транзакций небольшого объема.

Таким образом, выбор платформы для финансовых операций зависит от требуемого уровня безопасности, устойчивости и скорости транзакций. Платформы с высокой децентрализацией, такие как Ethereum, лучше подходят для открытых финансовых систем, тогда как приватные сети, такие как Hyperledger Fabric, предоставляют высокий уровень контроля и защиты для корпоративных пользователей. Ripple и Stellar предлагают компромисс между децентрализацией и скоростью, что делает их привлекательными для различных банковских и финансовых приложений.

Заключение

Проведенный сравнительный анализ блокчейн-платформ для финансовых транзакций показал, что выбор конкретной платформы зависит от требований к безопасности, скорости обработки транзакций, уровня децентрализации и энергопотребления. Каждая платформа обладает уникальными характеристиками, которые делают ее оптимальной для определенных задач. Ethereum, за счет своей децентрализации и смарт-контрактов, остается популярной для децентрализованных приложений и криптовалютных транзакций, в то время как Hyperledger Fabric и Ripple лучше подходят для корпоративного и межбанковского использования.

Параметры безопасности и устойчивости играют ключевую роль в выборе блокчейн-решений для финансовых операций. Децентрализованные платформы, такие как Ethereum, обеспечивают высокий уровень безопасности, однако требуют значительных ресурсов для поддержки своих алгоритмов. Платформы с централизованными и гибридными подходами, такие как Hyperledger Fabric и Ripple, обладают повышенной скоростью и масштабируемостью, что делает их удобными для коммерческого использования, где важны стабильность и контроль за доступом.

Таким образом, развитие и адаптация блокчейн-технологий в финансовой сфере продолжается, а с учетом текущих инноваций возможен рост числа платформ, поддерживающих эффективное управление данными, стабильные международные расчеты и масштабируемые финансовые сервисы. В дальнейшем можно ожидать создания новых решений, которые позволят оптимизировать финансовые транзакции, повышая безопасность и производительность.

Список литературы

1. Морозова Ю.А. Программные решения блокчейн в логистике и управлении цепями поставок // Информационное общество. 2019. №6. С. 49-58.
2. Сафиуллин М.Р., Шарифуллин М.Д., Ельшин Л.А. Перспективы использования блокчейн в системе организации международных цепочек поставок и трансграничных платежей // Региональная экономика и управление: электронный научный журнал. 2023. №4(76). С. 27.
3. Ларин О.Н., Буш Ю.Д., Некрутова С.П. Актуальные вопросы применения цифровых блокчейн-платформ для транспортной логистики // Интеллектуальный анализ данных и цифровая экономика. 2018. С. 8-22.
4. Сафиуллин М.Р., Абдукаева А.А., Ельшин Л.А., Савушкин М.В. Формализованная оценка сценарного развития национальной экономики в условиях проникновения блокчейн технологий в финансовый сектор // Вестник университета. 2020. №7. С. 154-162.

5. Финогеев А.Г., Васин С.М., Гамидуллаева Л.А., Финогеев А.А. Технология смарт контрактов на основе блокчейн для минимизации трансакционных издержек в региональных инновационных системах // Вопросы безопасности. 2018. №3. С. 34-55.
6. Сафиуллин М.Р., Бурганов Р.Т., Ельшин Л.А., Абдукаева А.А. Оценка влияния блокчейн технологий на национальную экономику: методические подходы и их апробация // Теоретическая и прикладная экономика. 2020. №3. С. 117-129.
7. Бахвалова Е.А., Судаков В.А. Исследование алгоритмов консенсуса для блокчейн-платформ // Препринты Института прикладной математики им. М.В. Келдыша РАН. 2021. С. 26-16.

УДК 004.7

ЭФФЕКТИВНОСТЬ ДЕЦЕНТРАЛИЗОВАННЫХ СЕТЕЙ В УПРАВЛЕНИИ ЛОГИСТИЧЕСКИМИ ЦЕПОЧКАМИ

Муромцев И.Л.
специалист, Московский физико-технический институт
(Москва, Россия)

EFFICIENCY OF DECENTRALIZED NETWORKS IN SUPPLY CHAIN MANAGEMENT

Muromtsev I.
specialist degree, Moscow Institute of Physics and Technology
(Moscow, Russia)

Аннотация

В статье представлен анализ возможностей использования децентрализованных сетей (ДС) в управлении логистическими цепочками. Рассмотрены ключевые преимущества технологии, такие как прозрачность, надежность и снижение транзакционных издержек. Приведены примеры успешного внедрения ДС крупными компаниями, включая Maersk и Walmart, что позволяет оценить влияние технологии на эффективность логистических процессов. ДС обеспечивают возможность обмена данными между участниками в режиме реального времени, что улучшает координацию и управление запасами. Также обсуждаются ограничения и вызовы, связанные с использованием ДС, такие как низкая скорость обработки данных и сложности интеграции с существующими системами. Ожидается, что дальнейшее развитие децентрализованных технологий будет способствовать их более широкому внедрению в логистические процессы, повышая прозрачность и безопасность цепочек поставок.

Ключевые слова: децентрализованные сети, логистика, блокчейн, управление цепочками поставок, прозрачность.

Abstract

This article provides an analysis of the potential of decentralized networks (DN) for supply chain management. Key benefits of this technology, including transparency, reliability, and reduced transaction costs, are discussed. Examples of successful DN implementation by large companies such as Maersk and Walmart are presented, illustrating the impact of this technology on logistics efficiency. DN enables real-time data sharing between participants, improving coordination and inventory management. The article also addresses limitations and challenges, such as low data processing speeds and integration difficulties with existing systems. It is anticipated that further development of decentralized technologies will promote their broader adoption in logistics processes, enhancing supply chain transparency and security.

Keywords: decentralized networks, logistics, blockchain, supply chain management, transparency.

Введение

С развитием глобальных цепочек поставок и увеличением сложности логистических операций все более важным становится поиск эффективных решений для управления логистическими цепочками. В последние годы внимание к децентрализованным сетям, таким как блокчейн, заметно возросло благодаря их уникальным возможностям по обеспечению

прозрачности и устойчивости к вмешательствам. Децентрализованные сети (ДС) позволяют участникам логистических цепочек обмениваться данными, управлять транзакциями и обеспечивать отслеживаемость товаров без необходимости в централизованных управляющих структурах. Основной целью данной статьи является исследование возможностей применения ДС в управлении логистическими цепочками и оценка их эффективности по ключевым параметрам.

Одной из основных проблем традиционных систем управления логистикой является отсутствие прозрачности и сложности в отслеживании передвижения товаров на всех этапах поставок. Это затрудняет контроль за выполнением условий и приводит к увеличению транзакционных затрат. Внедрение ДС позволяет решить эти задачи за счет создания единой сети, где каждый участник имеет доступ к актуальной информации в режиме реального времени. В данной статье будет рассмотрена эффективность использования ДС на различных этапах логистического процесса, включая управление запасами, транспортировку и контроль качества. Особое внимание уделено таким параметрам, как прозрачность, надежность и снижение транзакционных издержек.

Для достижения объективности в исследовании анализируются преимущества и ограничения децентрализованных сетей в сравнении с традиционными системами управления. Исследование также касается вопросов интеграции ДС в существующие логистические процессы и обсуждает возможные вызовы, связанные с использованием этой технологии, включая вопросы безопасности и защиты данных. Данная работа имеет целью предоставить целостное представление о применении ДС в логистике, выявить факторы, влияющие на их эффективность, и предложить рекомендации для дальнейшего использования.

Основная часть

Одним из главных преимуществ ДС для управления логистическими цепочками является обеспечение прозрачности и отслеживаемости на всех этапах поставок. В отличие от централизованных систем, где данные хранятся на серверах одного поставщика услуг, в ДС информация записывается в распределенный реестр, доступный каждому участнику сети. Это позволяет всем сторонам видеть актуальную информацию о местонахождении и статусе товаров в режиме реального времени, что существенно улучшает координацию и снижает риск недобросовестного поведения.

Еще одной ключевой особенностью ДС является снижение транзакционных издержек. Поскольку децентрализованные сети не требуют наличия централизованного посредника, управление данными происходит напрямую между участниками, что позволяет снизить затраты на верификацию и администрирование информации. ДС устраняют необходимость в дублирующих проверках, так как все данные доступны для всех участников и могут быть подтверждены автоматически [1]. Этот подход упрощает контроль за состоянием товаров, минимизируя количество ошибок и ускоряя выполнение операций.

Для повышения автоматизации в логистических операциях в рамках ДС активно применяются смарт-контракты – программируемые алгоритмы, встроенные в блокчейн. Смарт-контракты позволяют задавать и выполнять условия транзакций автоматически, без необходимости привлечения третьих лиц. Например, контракт может автоматически проводить оплату после подтверждения доставки товара. В случае выполнения условий смарт-контракт исполняется, что исключает человеческий фактор и минимизирует риск задержек. Таким образом, смарт-контракты позволяют оптимизировать многие процессы, от оплаты до контроля качества, что делает их особенно полезными в сложных логистических сетях.

Безопасность данных также является значимым преимуществом ДС. В централизованных системах информация подвержена риску взлома, в то время как в ДС данные защищены благодаря распределенной природе сети [2]. Каждое изменение должно быть подтверждено несколькими участниками, что затрудняет несанкционированные вмешательства и обеспечивает высокий уровень безопасности. Этот механизм делает ДС устойчивыми к атакам, что особенно важно для логистических цепочек, где любое нарушение целостности данных может повлечь за собой значительные финансовые потери.

Отслеживаемость и интеграция данных в ДС способствуют улучшению управления запасами. Возможность наблюдать за статусом товаров на всех этапах, начиная от отправки и заканчивая доставкой к конечному потребителю, позволяет более точно планировать пополнение запасов и минимизировать излишки. Это, в свою очередь, снижает издержки и повышает эффективность всей цепочки поставок. Кроме того, использование ДС позволяет значительно упростить процесс возврата товаров, поскольку система позволяет четко проследить их путь от потребителя к поставщику, что упрощает обработку и учет возвратов.

Хотя ДС предоставляют множество преимуществ, их использование сопровождается некоторыми ограничениями [3]. Одним из таких является относительно низкая скорость обработки данных по сравнению с централизованными системами, что ограничивает применение ДС в ситуациях, требующих высокой оперативности. Однако в условиях логистических цепочек, где важна надежность данных и возможность отслеживания, такие платформы находят широкое применение.

Примеры успешного внедрения децентрализованных сетей в логистические цепочки

В последние годы несколько крупных компаний и организаций успешно внедрили ДС для управления логистическими процессами, что помогло улучшить прозрачность, эффективность и надежность их цепочек поставок [4, 5].

1. Maersk и IBM (TradeLens)

Одним из наиболее известных примеров является проект TradeLens, созданный совместно компаниями Maersk и IBM. TradeLens использует блокчейн-платформу для управления грузовыми перевозками. Данная сеть позволяет участникам отслеживать информацию о статусе контейнеров в реальном времени, что значительно ускоряет процессы обработки документов и уменьшает вероятность ошибок и потерь данных. Благодаря этому, сроки доставки сократились, а затраты на административные процедуры снизились. TradeLens активно используют сотни компаний и портов, что делает эту платформу одним из примеров успешного применения ДС в глобальной логистике [6].

2. Walmart и IBM (Food Trust)

Компания Walmart внедрила блокчейн-платформу IBM Food Trust для отслеживания продуктов питания по всей цепочке поставок. Система позволяет точно определить источник и путь любого продукта от фермы до полки в магазине, что улучшает контроль качества и помогает быстрее выявлять проблемы, связанные с безопасностью продуктов. Благодаря Food Trust Walmart удалось сократить время, необходимое для отслеживания продуктов, с нескольких дней до нескольких секунд. Эта платформа активно используется в продуктовом секторе и другими крупными компаниями, такими как Nestlé и Carrefour, что демонстрирует потенциал ДС для повышения безопасности и прозрачности в пищевой индустрии [7].

3. DHL и Accenture

Логистический гигант DHL совместно с консалтинговой компанией Accenture внедрил блокчейн для отслеживания лекарственных препаратов. Проблемы, связанные с подделкой лекарств, стали серьезной угрозой в фармацевтической индустрии, и децентрализованная сеть позволяет отслеживать подлинность каждого препарата на всех этапах доставки. Система сохраняет записи о каждой партии лекарств и предоставляет информацию о местонахождении, что позволяет значительно снизить риск попадания поддельной продукции в розничную сеть. Этот пример подчеркивает роль ДС в повышении безопасности и прозрачности в медицинской и фармацевтической логистике [8].

4. Amazon и VeChain

Amazon использует блокчейн-платформу VeChain для улучшения контроля качества своих логистических цепочек. С помощью VeChain можно отслеживать происхождение и путь товаров, включая данные о хранении, температуре и условиях транспортировки. Это особенно актуально для товаров, требующих особых условий, таких как продукты питания или фармацевтическая продукция. ДС помогает Amazon повышать уровень доверия к поставщикам и минимизировать случаи потери или повреждения товаров [9].

5. Unilever и Provenance

Компания Unilever, стремясь к устойчивому развитию, использует платформу Provenance для отслеживания цепочек поставок сырья. Платформа позволяет фиксировать каждый этап производства и доставки, помогая Unilever обеспечить прозрачность и устойчивость своих процессов. Это помогает компании улучшить контроль за источниками сырья, включая продукты, выращенные с соблюдением экологических и социальных стандартов, что важно для ответственного подхода к производству и удовлетворения ожиданий потребителей [10].

Эти примеры показывают, что децентрализованные сети могут использоваться в различных отраслях, включая логистику, розничную торговлю, фармацевтику и пищевую промышленность. Внедрение ДС позволяет компаниям минимизировать транзакционные издержки, повысить прозрачность и отслеживаемость, улучшить контроль за качеством и укрепить доверие между участниками цепочек поставок.

Заключение

Внедрение ДС в логистические цепочки демонстрирует значительный потенциал для повышения прозрачности, надежности и эффективности управления поставками. Примеры успешного применения ДС крупными компаниями, такими как Maersk, Walmart, DHL и Amazon, показывают, что эта технология способна существенно улучшить контроль качества и отслеживаемость товаров, сократить транзакционные издержки и минимизировать риски, связанные с подделкой и утратой продукции. Это делает ДС актуальным решением для различных секторов, включая фармацевтику, продуктовую индустрию и розничную торговлю.

Несмотря на многочисленные преимущества, реализация децентрализованных сетей в логистике сопровождается рядом вызовов. Ограничения, такие как низкая скорость обработки транзакций и сложности интеграции с существующими системами, требуют дополнительных усилий и инвестиций. Тем не менее, перспективы развития и улучшения технологий, таких как смарт-контракты и совместимые стандарты, позволяют адаптировать ДС к потребностям крупных и малых предприятий, особенно в условиях глобализации поставок и повышенного внимания к устойчивому развитию.

В будущем можно ожидать, что децентрализованные сети будут все шире использоваться в логистике, предлагая новые инструменты для автоматизации и повышения безопасности логистических процессов. Эти технологии обеспечат более высокий уровень прозрачности и доверия между участниками цепочек поставок, способствуя устойчивому развитию и укреплению деловых отношений.

Список литературы

1. Ушаков М.А. Анализ инновационных методов и технологий в логистических цепочках предприятий // Организатор производства. 2023. Т. 31. №2. С. 109-124.
2. Бром А.Е., Терентьева З.С. Использование технологии блокчейн в управлении жизненным циклом продукции // Вестник Волжского университета им. В.Н. Татищева. 2018. Т. 2. №1. С. 118-124.
3. Красильников А.Б., Кочмашев О.Е. Проектирование системы логистического мониторинга на основе цепочки блоков (BLMS) // Системный анализ и аналитика. 2019. №2. С. 46-57.
4. Парфентьев Н.С. Возможности применения блокчейн-системы в портовой логистике // Развитие современной экономики России. 2022. С. 222-227.
5. Сергеев В.И., Кокурин Д.И. Применение инновационной технологии «Блокчейн» в логистике и управлении цепями поставок // Креативная экономика. 2018. Т. 12. №2. С. 125-140.
6. Башарова Э.И., Веселова М.П., Татаева И.Ю. Цифровизация логистических процессов на основе опыта компаний" Maersk" и" IBM" // Modern Science. 2019. №11-2. С. 40-44.
7. Tan B., Yan J., Chen S., Liu X. The impact of blockchain on food supply chain: The case of walmart //Smart Blockchain: First International Conference, SmartBlock 2018, Tokyo, Japan, December 10–12, 2018, Proceedings 1. Springer International Publishing. 2018. P. 167-177.
8. Якубанец С. Блокчейн в логистике: движение вперед // Логистика. 2018. №6. С. 12-15.

9. She Z. Vechain: A renovation of supply chain management—A look into its organization, current activity, and prospect // Proceedings of the 2022 International Conference on Educational Informatization, E-commerce and Information System, Macao, China. 2022. P. 29-30.
10. Мухамедова З.Г., Осадчук В.Д., Тулаев А.У. Перспективы использования технологии блокчейн в организации перевозочного процесса и цепочке поставок // Известия Транссиба. 2022. №2(50). С. 142-156.

UDC 004.4

EFFICIENCY OF CONTAINERIZATION IN ORGANIZING INFRASTRUCTURE FOR IT PROJECTS

Rudenskaya O.
master's degree, Peter the Great St. Petersburg Polytechnic University (St. Petersburg, Russia)

ЭФФЕКТИВНОСТЬ КОНТЕЙНЕРИЗАЦИИ В ОРГАНИЗАЦИИ ИНФРАСТРУКТУРЫ IT-ПРОЕКТОВ

Руденская О.Ю.
магистр, Санкт-Петербургский политехнический университет Петра Великого (Санкт-Петербург, Россия)

Abstract

This article explores the impact of containerization on IT project infrastructure, analyzing its benefits and limitations. Containerization simplifies dependency management, enhances flexibility, and speeds up deployment by isolating environments. Examples from major companies such as Netflix, Spotify, and Google highlight its advantages in scaling and adapting infrastructure to high loads. Special attention is given to container orchestration using Kubernetes, which facilitates managing microservices architecture. The limitations of containerization, including the need for improved security and data management, are also discussed. It is anticipated that containerization will become an integral part of IT infrastructure, especially within DevOps environments.

Keywords: containerization, IT infrastructure, Docker, Kubernetes, microservices.

Аннотация

В статье рассмотрено влияние контейнеризации на организацию инфраструктуры IT-проектов, анализируются её преимущества и ограничения. Контейнеризация позволяет упрощать управление зависимостями приложений, повышать гибкость и скорость развертывания за счёт изоляции окружений. Примеры использования контейнеризации в крупных компаниях, таких как Netflix, Spotify и Google, демонстрируют её преимущества в улучшении масштабируемости и адаптации инфраструктуры к высоким нагрузкам. Особое внимание уделено вопросам оркестрации контейнеров с использованием Kubernetes, что способствует упрощению управления микросервисной архитектурой. Также обсуждаются ограничения контейнеризации, включая потребность в усиленной безопасности и контроле над данными. Ожидается, что контейнеризация станет неотъемлемой частью IT-инфраструктуры, особенно в DevOps-средах.

Ключевые слова: контейнеризация, инфраструктура IT, Docker, Kubernetes, микросервисы.

Introduction

The modern development of information technology and the widespread adoption of cloud computing have necessitated innovative approaches to IT project infrastructure management. Containerization, a method of packaging applications and their dependencies into containers, has enhanced resource management flexibility and improved application deployment. Containers allow applications to run in an isolated environment, minimizing dependency conflicts and simplifying portability. This article aims to study the impact of containerization on IT infrastructure organization and assess its effectiveness in various aspects.

One of the main challenges of traditional IT infrastructure is the high dependency of applications on specific operating environments, which complicates deployment in diverse settings and increases infrastructure management costs. Containerization addresses these issues by providing a universal approach that allows containers to run on any server supporting container platforms such as Docker or Kubernetes. This article analyzes key aspects of containerization, including portability, scalability, and resource management, which directly impact the efficiency of IT project infrastructure.

Despite its advantages, containerization also faces limitations related to security, data management, and container network control. This study addresses these aspects and offers recommendations for minimizing potential risks when implementing containerized infrastructure. Summarizing current approaches and methods used to manage containerized infrastructure will present a comprehensive view of its application in IT projects and assess its contribution to improving infrastructure flexibility and efficiency.

Main part

Containerization has become one of the key approaches in organizing IT project infrastructure, as it provides high flexibility and application portability [1]. Figure 1 illustrates the growth of container adoption over time, reflecting the increasing popularity of containerization in IT infrastructure management across various industries.

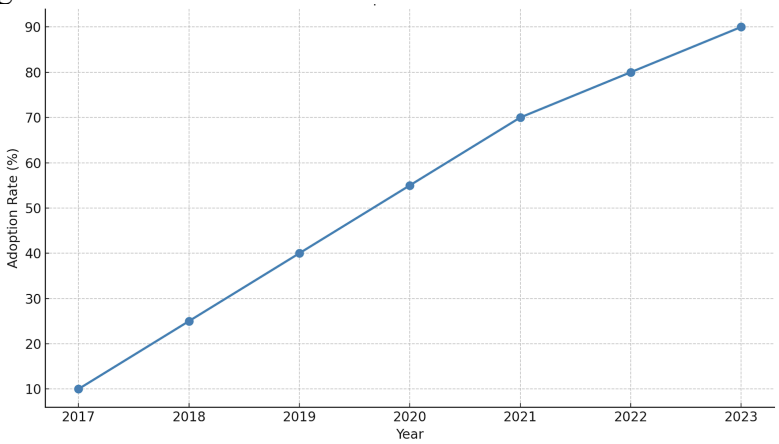

Figure 1. Container adoption growth over time

As shown in Figure 1, the adoption rate of containerization has significantly increased over recent years, rising from 10% in 2017 to 90% in 2023. This rapid growth highlights the effectiveness of container technologies in improving application scalability, portability, and deployment speed. The steady upward trend also indicates widespread recognition of containers as a vital component of modern IT infrastructure. The increasing adoption underscores the importance of platforms like Kubernetes, which facilitate the management and orchestration of containerized applications. Containers are lightweight, isolated environments that can start much faster than virtual machines (VMs) due to the lack of a full operating system load. This enhances application adaptability across different environments and reduces infrastructure maintenance costs.

Kubernetes is widely used for container management and scaling, automating container orchestration, and enabling flexible management across various environments. Kubernetes includes features for load distribution, network connection management, and data storage [2]. Figure 2 shows an architecture diagram of Kubernetes, featuring containers deployed in pods, enabling application management and process automation.

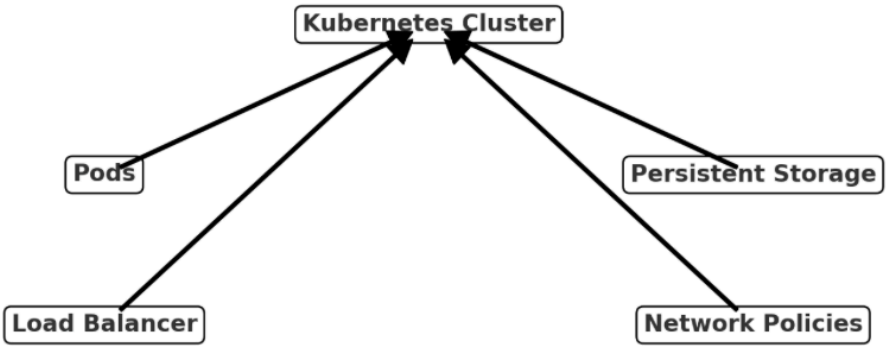

Figure 2. Kubernetes architecture

In addition to accelerating deployment processes, containerization facilitates DevOps practices, including integration and deployment (CI/CD). Containers allow creating uniform environments, making the testing process more precise and avoiding issues related to incompatibility between test and production environments. This is especially important for large IT projects with frequent updates, where application stability and reduced development time are crucial [3].

Securing a containerized infrastructure requires a special approach, as containers have a lower isolation level than VMs. To protect container environments, Kubernetes network policies and an access role system are used, limiting data access and managing confidential information. For example, applying network policies in Kubernetes can control interactions between containers, preventing unauthorized data access. However, for critical applications, additional security measures, such as specialized containers with enhanced isolation, may be necessary.

Data management within containers is also essential [4]. Since containers are isolated and temporary by nature, any data stored within them may be lost when they are terminated. In such cases, persistent storage solutions like Persistent Volumes in Kubernetes are used, allowing data to be stored outside the container and ensuring its availability even after restarts. This is particularly important for applications with high data retention and infrastructure reliability requirements.

Examples of containerization in real projects

Containerization has already proven effective in several large IT companies that have implemented it to optimize infrastructure and increase the flexibility of their solutions. This section presents successful examples of container usage in real IT projects, highlighting the practical value of this technology.

One prominent example is Netflix, which uses containers to manage scalable microservices. Netflix's streaming platform requires high scalability and availability, as its services must handle a vast number of real-time requests. Containerization has allowed Netflix to standardize and speed up application deployment, using containers for individual functional modules, such as recommendations, user data processing, and content loading. Implementing containers significantly reduced deployment and testing time, improving overall infrastructure performance [5].

Another example is Spotify, which employs containerization to support its music services and ensure seamless application operation. Spotify uses containers to organize a microservice architecture, where each container is responsible for a specific function, such as streaming, playlist management, and analytics. This structure allows Spotify to update individual modules quickly without affecting the entire system. Additionally, containers provide a unified environment for testing and production, avoiding issues related to version and environment incompatibilities.

Google also uses containers in its internal infrastructure to optimize resource utilization and improve manageability. Google actively uses Kubernetes for container orchestration and distributed load management. Containerization helps Google maintain high reliability in services like Google Search and Gmail, where stability and high performance are crucial due to heavy traffic volumes [6]. Moreover, Kubernetes allows Google to scale its infrastructure based on load, reducing maintenance costs.

In e-commerce, containerization is utilized by Alibaba, which uses containers to manage infrastructure and support peak loads during major sales events. Containerization enables Alibaba to flexibly scale its server capacity during traffic surges, such as "Singles' Day." This ensures platform stability and high performance even under increased demand [7-9].

These examples demonstrate that containerization provides companies with the flexibility to manage infrastructure, enhancing its resilience and reducing application deployment time. Implementing containers improves performance, simplifies application management, and ensures high availability for end users.

Conclusion

Implementing containerization in IT project infrastructure organization significantly improves application flexibility, scalability, and resilience, which is especially important given today's high demands for rapid deployment and service availability. Examples from major companies like Netflix, Spotify, and Google show that containerization not only simplifies microservice management and reduces deployment time but also promotes optimal resource utilization, reducing operational costs and enhancing performance.

Using the Kubernetes platform, widely adopted for container orchestration, allows companies to manage resources flexibly and adapt infrastructure to current loads. This not only ensures resilience during peak demand periods but also minimizes downtime, essential for mission-critical services operating in real time. Meanwhile, containerization requires careful attention to security, data management, and access control.

In the future, containerization's popularity is expected to grow, especially in DevOps and CI/CD practices. Container technologies will evolve, offering improved methods for automation and data protection, enabling companies to leverage this technology more effectively to achieve business goals and enhance service quality.

References

1. Basharimova M.V., Podluzhny V.S. Improving the efficiency of cross-functional team formation in IT projects // International Scientific and Technical Conference of Young Scientists of BSTU named after V.G. Shukhov, dedicated to the 300th anniversary of the Russian Academy of Sciences. 2022. P. 51.
2. Serdechny A.L. Cyberspace as a subject of research and protection. Part 1 // Information and Security. 2021. Vol. 24. No.3. P. 309-326.
3. Binenda A.D. Practical application of the Doc as Code approach in IT companies using the Antora tool // Current Research. 2024. No.19 (201). P. 17-20.
4. Shestakov K.I., Sokolov I.M., Pirogov M.A., Soloviev S.G. Experience in the development, implementation, and standardization of BIM (TIM) design in the mining industry // Mining Industry. 2021. No.5. P. 40.
5. Bakhtizin V.V., Neborsky S.N. Creating a managed software architecture // Software Products and Systems. 2006. No.3. P. 2-5.
6. Kurganova N.V., Filin M.A., Chernyaev D.S., Shaklein A.G., Namyot D.E. Implementation of digital twins as a key direction of production digitalization // International Journal of Open Information Technologies. 2019. Vol. 7. No.5. P. 105-115.
7. Doroshenko A.N., Tkachev L.L. On the benefits of discrete modeling of real systems // Innovations in Industries as a Factor in Solving Socio-Economic Problems of Modernity. 2012. P. 81-95.
8. Casalicchio E., Iannucci S. The state-of-the-art in container technologies: Application, orchestration and security // Concurrency and Computation: Practice and Experience. 2020. Vol. 32. No.17. P. e5668.
9. Notteboom T., Rodrigue J.P. The future of containerization: perspectives from maritime and inland freight distribution // GeoJournal. 2009. Vol. 74. P. 7-22.

UDC 004.6

ANALYSIS OF EFFICIENCY IN STORING AND PROCESSING UNSTRUCTURED DATA IN BIG DATA ENVIRONMENTS

Aliyev D.
bachelor's degree, Azerbaijan State University of Economics (Baku, Azerbaijan)

АНАЛИЗ ЭФФЕКТИВНОСТИ ХРАНЕНИЯ И ОБРАБОТКИ НЕСТРУКТУРИРОВАННЫХ ДАННЫХ В СРЕДЕ BIG DATA

Алиев Д.В.
бакалавр, Азербайджанский государственный экономический университет (Баку, Азербайджан)

Abstract

The article examines key technologies for storing and processing unstructured data within Big Data environments, including Hadoop, NoSQL databases, Apache Spark, and Elasticsearch. Key advantages and limitations of each approach, along with their impact on infrastructure performance and scalability, are analyzed. An example is provided using Apache Kafka for data streaming and PySpark for preprocessing, highlighting the significance of these technologies in handling large volumes of information. Recommendations are given for selecting suitable technologies for different business scenarios. The research demonstrates that the integration of Big Data technologies into business processes enhances flexibility and reduces costs associated with data storage and processing.

Keywords: unstructured data, big data, containerization, data analysis, Apache Spark, Elasticsearch.

Аннотация

В статье рассмотрены основные технологии для хранения и обработки неструктурированных данных в среде больших данных (Big Data), включая Hadoop, NoSQL базы данных, Apache Spark и Elasticsearch. Проанализированы ключевые преимущества и ограничения каждого подхода, а также их влияние на производительность и масштабируемость инфраструктуры. Приведен пример использования Apache Kafka для потоковой передачи данных и PySpark для предобработки, что подчеркивает важность этих технологий в управлении большими объемами информации. Даны рекомендации по выбору подходящих технологий для различных бизнес-сценариев. Исследование показывает, что интеграция Big Data технологий в бизнес-процессы повышает гибкость и снижает затраты на хранение и обработку данных.

Ключевые слова: неструктурированные данные, большие данные, контейнеризация, анализ данных, Apache Spark, Elasticsearch.

Introduction

With the development of data processing technologies and the increase in data volume, the need for efficient solutions for data storage and analysis is growing. One of the key challenges modern organizations faces is the need to work with unstructured data, which constitutes a significant portion of information coming from various sources. Unstructured data, such as text documents, images, videos, and social media data, cannot be easily processed by traditional relational databases. Consequently, there is a need to develop technologies and methodologies capable of effectively

storing and analyzing such data. The goal of this article is to explore existing approaches to the storage and processing of unstructured data in Big Data environments.

One of the main challenges of working with unstructured data is its complexity and diversity. Unlike structured data, which is easily systematized, unstructured data requires more flexible storage and processing approaches. In recent years, technologies such as Hadoop and NoSQL databases have been actively used in Big Data environments, providing distributed data storage and enabling data processing under high loads. This article presents a comparative analysis of technologies used for the storage and processing of unstructured data and evaluates their effectiveness based on key parameters such as processing speed, scalability, and reliability.

In addition to technical aspects, special attention is given to optimizing processes for working with unstructured data and reducing processing costs. The introduction of technologies such as Apache Spark and Elasticsearch provides opportunities for fast data processing and analysis, which is particularly relevant in the context of modern digital business. This article examines the prospects and limitations of these technologies depending on the data characteristics and business needs and offers recommendations for selecting suitable solutions for different usage scenarios.

Main part

One of the primary methods for storing unstructured data is the use of distributed file systems, such as the Hadoop Distributed File System (HDFS). HDFS enables storing large volumes of data by dividing it into blocks and distributing them across multiple nodes, which enhances storage reliability and protects data from loss [1]. For processing unstructured data based on HDFS, Apache Spark is often used – a framework that enables parallel computations and speeds up the analysis process. Unlike traditional MapReduce, Spark offers higher performance by utilizing memory, making it suitable for tasks requiring fast data processing.

NoSQL databases, such as MongoDB and Cassandra, have also become popular tools for storing and processing unstructured data. Unlike relational databases, NoSQL databases can handle various data types, such as JSON documents and graphs, making them flexible for Big Data projects [2]. MongoDB, for instance, allows storing data in a document format, which simplifies working with text information and enables quick search and filtering operations. On the other hand, Cassandra ensures high availability and scalability, making it suitable for applications with heavy loads [3].

Query optimization and reducing data processing time are essential aspects of working with unstructured data. Elasticsearch, a search system built on Apache Lucene, is used for indexing and searching through large volumes of unstructured information. Elasticsearch allows creating indexes for text data and provides high search speed, which is particularly important in real-time data analysis. Table 1 presents the main characteristics of Hadoop, MongoDB, and Elasticsearch, including their applicability to different data types and performance.

Table 1

Comparison of technologies for unstructured data storage and processing

Technology	**Data type**	**Advantages**	**Limitations**	**Common application areas**
Hadoop	All types	Scalability, reliability	Requires significant resources	Big Data analysis, long-term storage
MongoDB	Documents (JSON)	Flexibility, easy integration	Limited ACID support	Document storage, data analytics
Elasticsearch	Text data	High search speed	Dependent on indexing	Search systems, text analytics
Cassandra	All types	High availability, scalability	Complex setup, resource-intensive	Analytics, high-load systems

Apache Spark	All types	High performance, memory support	Memory-dependent	Data analysis, fast processing of large volumes

To improve the efficiency of unstructured data processing, technologies such as Apache Kafka play a key role, providing real-time data transfer between different system components [4]. Kafka serves as a distributed streaming service that transmits data from sources to processors, allowing rapid processing and analysis of unstructured data. This approach is used for real-time analytics, including monitoring and event tracking, where minimal latency is essential.

An important aspect of unstructured data processing is ensuring data security and integrity. Most technologies, such as MongoDB and Elasticsearch, support built-in data protection mechanisms, including access control and encryption. However, for Big Data containing unstructured information, additional measures, such as access rights distribution and visibility restrictions, are often required, especially in corporate and government projects [5].

Data quality control is also essential, as unstructured data may contain errors and incorrect values. Data quality analysis tools, such as Apache Griffin, help identify anomalies and ensure data meets quality standards. This is especially important when integrating data from various sources, where each error can impact the final analysis results.

Applying Big Data technologies in unstructured data management

Several key technologies are actively applied in Big Data environments for working with unstructured data, each fulfilling a specific role in the data processing and analysis process. These technologies include distributed data storage systems such as Hadoop and NoSQL databases, data streaming tools like Apache Kafka, and search systems such as Elasticsearch.

The first stage of managing unstructured data is collecting and integrating it from various sources [6]. In this task, solutions like Apache Kafka, which processes data in real time and transmits it to storage and analytical systems, become indispensable. Kafka supports low-latency data transmission, which is crucial for applications where data analysis must be immediate. For streaming data processing during analysis stages, Kafka can be integrated with Apache Spark, providing high performance.

Once data is collected, it must be stored and prepared for further processing. Hadoop Distributed File System (HDFS) offers a reliable and scalable solution for storing large volumes of unstructured data. Data is distributed across multiple nodes, ensuring high fault tolerance and data security. HDFS is widely used for long-term data storage and allows processing data using tools such as MapReduce and Spark [7].

NoSQL databases, such as MongoDB and Cassandra, are widely used for effectively storing textual and semi-structured information. These databases allow data to be stored in flexible formats, such as JSON, which is especially useful for document and text information processing. MongoDB provides fast data search and filtering, while Cassandra is optimized for distributed systems and can handle low-latency queries, making it suitable for high-load applications.

When data is ready for analysis, tools like Apache Spark and Elasticsearch can be used for processing. Spark is a high-performance framework for Big Data processing that supports parallel computations in clusters, allowing data analysis at high speed. In turn, Elasticsearch allows creating indexes for text information, speeding up the search process and making it effective even with large data volumes [8]. Elasticsearch is widely used for text search and real-time data analysis, making it suitable for applications that require quick response.

For example, let’s add a Python code snippet that shows how to use the PySpark library (Python API for Apache Spark) and Elasticsearch for unstructured data processing and analysis. In this code, we assume we have text data, which will first be processed with PySpark and then indexed and stored in Elasticsearch for fast search and analysis.

```
# Importing PySpark and Elasticsearch libraries
from pyspark.sql import SparkSession
from pyspark.sql.functions import col, lower, regexp_replace
from elasticsearch import Elasticsearch, helpers
```

```
# Initializing a Spark session
spark = SparkSession.builder \
    .appName("Big Data Processing Example") \
    .getOrCreate()

# Reading unstructured data from a text file
data_path = "path/to/your/textfile.txt"
df = spark.read.text(data_path)

# Data preprocessing: cleaning text and converting to lowercase
df_cleaned = df.withColumn("value", lower(col("value")))
df_cleaned = df_cleaned.withColumn("value", regexp_replace(col("value"), "[^a-zA-Z0-9\\s]", ""))

# Transforming data to a structure suitable for analysis
df_transformed = df_cleaned.withColumnRenamed("value", "processed_text")

# Converting DataFrame to list for loading into Elasticsearch
processed_data = df_transformed.rdd.map(lambda row: row["processed_text"]).collect()

# Initializing Elasticsearch connection
es = Elasticsearch("http://localhost:9200")

# Function to prepare data in Elasticsearch format
def es_data_generator(data, index_name="processed_data_index"):
    for line in data:
        yield {
            "_index": index_name,
            "_source": {
                "text": line
            }
        }

# Loading data into Elasticsearch
helpers.bulk(es, es_data_generator(processed_data))

print("Data successfully processed and indexed in Elasticsearch.")

# Stopping the Spark session
spark.stop()
```

This example demonstrates the process of unstructured data processing with PySpark and its subsequent loading into Elasticsearch. First, a Spark session is created for loading and cleaning text data. After preprocessing, which includes removing special characters and converting text to lowercase, the data is transformed into a structure suitable for analysis. Then it is converted to a format convenient for loading into Elasticsearch. In Elasticsearch, the data is indexed, allowing for fast search and real-time text information analysis.

Conclusion

Containerization and Big Data technologies represent a significant breakthrough in managing and processing unstructured data, allowing organizations to handle the increasing volumes of information and meet modern business demands. The use of Hadoop, Apache Spark, NoSQL databases, and Elasticsearch substantially enhances the flexibility and performance of infrastructure

for storing and analyzing data. These technologies have proven effective under high loads, providing capabilities for fast processing and real-time data analysis.

Furthermore, the use of data streaming tools such as Apache Kafka enables flexibility and timeliness in processing incoming real-time data. Although additional measures are required for ensuring data security and quality control, the application of containerization and Big Data technologies allows organizations to reduce operational costs and improve the reliability of information systems.

Future developments are expected to further advance these technologies and integrate them with new machine learning methods, establishing a foundation for enhanced capabilities in intelligent analysis and processing of unstructured information. This ongoing progress will open up new possibilities for more sophisticated data handling, ensuring that businesses can leverage unstructured data as a valuable resource in achieving their goals and improving service quality.

References

1. Gordienko E.P., Panenko N.S. Modern technologies for processing and analyzing Big Data in scientific research // Current Problems of Railway Transport. 2018. P. 44-48.
2. Mamedova G.A., Zeynalova L.A., Melikova R.T. Big Data technologies in e-education // Open Education. 2017. Vol. 21. No.6. P. 41-48.
3. Stepanova D.I. The use of the BIG DATA system to improve the efficiency of utilities // World Economy: Security Issues. 2019. No.2. P. 70-78.
4. Fedorova L.A., Hu G., Huang S., Zemlyakova S.A. The application of Big Data technologies in modern enterprises // Bulletin of the Altai Academy of Economics and Law. 2020. No.9-2. P. 322-329.
5. Kulikova O.M., Tropynina N.E. Challenges of using BIG DATA technology in modern market conditions // Innovative Economy: Development Prospects and Improvement. 2022. No.7 (65). P. 16-21.
6. Radchenko I.A., Nikolaev I.N. Big Data technologies and infrastructure // St. Petersburg: ITMO University. 2018. Vol. 52.
7. Alekseev K.A. The use of Big Data in international business // Proceedings of the Institute for System Programming RAS. 2020. Vol. 32. No.4. P. 7-20.
8. Gladchenko V.A. The use of global «Big Data» technologies as an effective tool for risk management in customs authorities // Economics. Law. Innovation. 2019. No.2. P. 36-41.

УДК 004.9

РАЗРАБОТКА ИНТЕЛЛЕКТУАЛЬНЫХ СИСТЕМ МОНИТОРИНГА ДЛЯ УМНЫХ ГОРОДОВ

Суворова К.В.
Дальневосточный федеральный университет
(Владивосток, Россия)

DEVELOPMENT OF INTELLIGENT MONITORING SYSTEMS FOR SMART CITIES

Suvorova K.
Far Eastern Federal University
(Vladivostok, Russia)

Аннотация

Интеллектуальные системы мониторинга играют ключевую роль в развитии умных городов, обеспечивая сбор и анализ данных, необходимых для оперативного управления городской инфраструктурой. Основная цель данной статьи – исследование современных подходов к разработке интеллектуальных систем мониторинга и оценка их перспективных возможностей для использования в городской среде. Рассматриваются вопросы, связанные с интеграцией и обработкой данных в реальном времени, а также применение технологий машинного обучения и больших данных для автоматизации процессов управления. В статье также анализируются основные вызовы, стоящие перед внедрением данных систем, такие как безопасность данных, затраты на инфраструктуру и соблюдение правовых норм. Рассмотренные подходы и методы позволяют выделить важные факторы для успешного внедрения интеллектуальных систем мониторинга и их использования в городах, ориентированных на устойчивое развитие.

Ключевые слова: интеллектуальные системы мониторинга, умные города, обработка данных, городская инфраструктура, большие данные.

Abstract

Intelligent monitoring systems play a pivotal role in the development of smart cities by providing data collection and analysis necessary for efficient urban infrastructure management. The primary goal of this article is to examine modern approaches to designing intelligent monitoring systems and to assess their potential for urban use. Issues related to real-time data integration and processing are explored, as well as the application of machine learning and big data technologies for process automation. The article also discusses major challenges in implementing these systems, such as data security, infrastructure costs, and legal compliance. The reviewed approaches and methods highlight key factors for successful deployment of intelligent monitoring systems in cities aiming for sustainable development.

Keywords: intelligent monitoring systems, smart cities, data processing, urban infrastructure, big data.

Введение

Современные города сталкиваются с быстро растущими объемами данных, генерируемых различными источниками, включая транспортные системы, коммунальные службы, общественные места и частные устройства пользователей [1]. В условиях быстрого роста населения и увеличения уровня урбанизации возникает необходимость в эффективных

системах управления, которые обеспечивают поддержку решений для улучшения качества жизни горожан. Одним из перспективных направлений является внедрение интеллектуальных систем мониторинга, основанных на обработке и анализе данных, которые позволяют решать задачи планирования, безопасности, экологии и управления городской инфраструктурой. Целью данной статьи является исследование подходов к разработке интеллектуальных систем мониторинга, анализ их ключевых компонентов и перспектив применения для умных городов.

Одной из главных задач, которую решают интеллектуальные системы мониторинга, является интеграция и обработка данных в режиме реального времени. В умных городах используется множество сенсоров, датчиков и систем видеонаблюдения, которые собирают огромные массивы информации. Эти данные нуждаются в оперативной обработке и анализе для обеспечения точного и своевременного реагирования на события и для поддержки принятия решений. Для решения данных задач активно применяются технологии искусственного интеллекта, машинного обучения и Big Data, которые предоставляют возможности для автоматизации и повышения точности предсказаний. В рамках данной статьи будет рассмотрена структура и функциональные возможности интеллектуальных систем мониторинга, ориентированных на задачи анализа данных и поддержки решений для умных городов [2].

Создание эффективных систем мониторинга требует учета множества факторов, включая безопасность данных, отказоустойчивость и масштабируемость инфраструктуры. Кроме того, важно учитывать специфику городской среды, где данные могут поступать из различных и независимых источников, что приводит к необходимости использования распределённых вычислений и гибридных архитектур. В данной статье проведен обзор основных архитектурных решений для построения интеллектуальных систем мониторинга, а также анализируются их достоинства и недостатки в зависимости от различных задач и условий эксплуатации в городской инфраструктуре. Предложенные методы и технологии способны поддерживать устойчивое развитие городов и оптимизировать управление городской средой.

Основная часть

Одним из ключевых элементов интеллектуальных систем мониторинга является платформа для обработки данных, которая включает модули сбора, хранения и анализа информации. Эти модули обеспечивают сбор данных из различных источников, таких как сенсоры, камеры наблюдения и интернет-устройства (IoT), и передают их в хранилище для дальнейшей обработки. В умных городах активно применяются распределенные системы, такие как Hadoop и Apache Spark, которые позволяют обрабатывать большие объемы данных в реальном времени. Для анализа данных в режиме реального времени используются технологии машинного обучения, которые позволяют выявлять аномалии, предсказывать события и поддерживать управление городской инфраструктурой. Примером может служить использование алгоритмов кластеризации для анализа транспортных потоков. Данные о движении транспорта поступают в систему, где на основе моделей машинного обучения определяются аномалии, такие как пробки или дорожные происшествия. Этот подход позволяет своевременно реагировать на события и перенаправлять транспортные потоки для оптимизации движения [3].

Важной составляющей интеллектуальных систем мониторинга является работа с текстовыми и визуальными данными. Например, для анализа информации из социальных сетей или сообщений от горожан может быть полезен метод обработки естественного языка (NLP), который позволяет анализировать содержание сообщений и выделять значимые события. Таким образом, система может выявлять негативные настроения или жалобы, связанные с городскими услугами, и передавать их соответствующим службам для реагирования. Это позволяет оперативно учитывать мнение населения и улучшать качество услуг [4].

Для эффективного управления городскими службами применяется также анализ данных о потреблении ресурсов, таких как электроэнергия, вода и тепло. На основе данных сенсоров система мониторинга может обнаруживать нерациональное использование ресурсов, а также

выявлять потенциальные аварии, например, утечки воды или неисправности в электрораспределительных сетях. Системы мониторинга могут взаимодействовать с коммунальными службами и автоматически направлять запросы на ремонт или модернизацию оборудования.

Для визуализации данных и удобства их анализа в интеллектуальных системах мониторинга используется интеграция с геоинформационными системами, что позволяет представлять данные в виде карт, диаграмм и графиков [5]. Например, информация о загрязнении воздуха, поступающая от сенсоров, может отображаться на интерактивной карте, где разные районы города отмечены в зависимости от уровня загрязнения.

Для примера добавим фрагмент кода на Python с использованием библиотеки folium для визуализации данных о состоянии транспорта в реальном времени на карте. В данном примере используются координаты GPS для отображения текущего местоположения общественного транспорта.

```
import folium

# Инициализация карты
m = folium.Map(location=[55.751244, 37.618423], zoom_start=12)  # Москва, Россия

# Пример данных с координатами транспорта
transport_data = [
    {"name": "Bus 123", "location": [55.760, 37.620]},
    {"name": "Tram 7", "location": [55.755, 37.615]},
    {"name": "Metro Line 5", "location": [55.749, 37.622]}
]

# Добавление маркеров на карту
for vehicle in transport_data:
    folium.Marker(location=vehicle["location"], popup=vehicle["name"]).add_to(m)

# Сохранение карты в HTML файл
m.save("transport_map.html")
```

Этот код создает интерактивную карту, на которой отображаются транспортные средства с указанием их положения. Такой подход позволяет визуализировать информацию о передвижении транспорта, что полезно для управления транспортными потоками в городе.

Проблемы внедрения интеллектуальных систем мониторинга для умных городов

Внедрение интеллектуальных систем мониторинга в умные города, несмотря на очевидные преимущества, сталкивается с рядом сложных задач и вызовов. Эти проблемы варьируются от технических трудностей и вопросов кибербезопасности до финансовых и правовых ограничений, затрудняющих процесс реализации данных систем в городской инфраструктуре [6-8].

Первой ключевой проблемой является безопасность и защита данных. Поскольку интеллектуальные системы собирают и обрабатывают большие объемы информации о городских процессах и жителях, существует высокий риск утечки данных или их несанкционированного использования. Данные, собираемые сенсорами и IoT-устройствами, могут содержать конфиденциальную информацию о перемещениях людей, их активности и даже персональные данные. Защита этих данных требует внедрения сильных механизмов шифрования и контроля доступа, что может значительно усложнить и удорожить систему. Безопасность также затруднена из-за разнородности оборудования и программного обеспечения, что создает дополнительные уязвимости.

Второй значительной проблемой является вопрос масштабируемости и интеграции. Современные города используют разнородные системы и технологии, которые зачастую несовместимы друг с другом. Это создает трудности при интеграции интеллектуальных систем

с уже существующими системами управления и требует разработки унифицированных стандартов и протоколов. Масштабируемость системы также является критическим аспектом, поскольку объем данных, обрабатываемых в умных городах, постоянно растет. Без возможности масштабирования системы городские платформы мониторинга могут столкнуться с перебоями в работе и потерей данных при увеличении количества подключённых устройств и источников информации [9].

Третьим вызовом является высокая стоимость внедрения и эксплуатации интеллектуальных систем. Умные системы мониторинга требуют значительных инвестиций в инфраструктуру, оборудование и программное обеспечение. Помимо начальных затрат на покупку и установку, также требуются постоянные расходы на обслуживание, обновление и модернизацию оборудования и программ. Финансирование таких систем может стать проблемой, особенно для небольших городов с ограниченным бюджетом. В некоторых случаях для поддержки внедрения интеллектуальных систем могут потребоваться субсидии или гранты, что увеличивает зависимость от внешнего финансирования.

Четвертая проблема касается правовых и этических аспектов. Внедрение интеллектуальных систем неизбежно связано с использованием и обработкой данных о жителях города, что вызывает вопросы конфиденциальности и соблюдения законодательства. Необходимо соблюдать нормы, регулирующие защиту персональных данных, такие как GDPR в Европе. Кроме того, некоторые жители могут воспринимать системы мониторинга как вторжение в их частную жизнь, что требует особого подхода к информированию и получению согласия пользователей на сбор данных [10]. Таким образом, создание и поддержка прозрачности в сборе и использовании данных становится важной задачей для управления городской инфраструктурой.

Пятая проблема связана с устойчивостью и надежностью системы. В условиях умного города любые сбои в работе интеллектуальной системы мониторинга могут привести к значительным сбоям в управлении городской инфраструктурой. Например, отказ системы мониторинга движения может вызвать сбой в транспортных потоках, что повлияет на весь город. Устойчивость и отказоустойчивость системы являются обязательными требованиями для успешного внедрения интеллектуальных решений, однако их обеспечение требует дополнительных ресурсов и создания резервных систем, что также повышает затраты [11].

Заключение

Внедрение интеллектуальных систем мониторинга в умные города предоставляет широкие возможности для повышения качества жизни горожан, улучшения управления городской инфраструктурой и оптимизации использования ресурсов. Такие системы обеспечивают непрерывный сбор и анализ данных из различных источников, что позволяет оперативно реагировать на события и принимать обоснованные решения. В статье был проведен анализ ключевых компонентов и технологий, лежащих в основе интеллектуальных систем мониторинга, а также рассмотрены их возможности для управления транспортными потоками, экологическим состоянием и другими аспектами городской среды.

Несмотря на многочисленные преимущества, внедрение интеллектуальных систем связано с рядом значительных вызовов, таких как безопасность данных, высокие финансовые затраты, сложность интеграции с существующими системами и правовые ограничения. Эти проблемы требуют внимательного подхода, особенно при работе с большими объемами информации, которая может включать персональные данные жителей. Разработка стандартизированных протоколов, обеспечение масштабируемости инфраструктуры и реализация эффективных мер защиты данных представляют собой важные задачи для успешного применения данных технологий.

Таким образом, успешное внедрение интеллектуальных систем мониторинга в умные города требует комплексного подхода, включающего технические, правовые и финансовые аспекты. Умные города, опираясь на интеллектуальные технологии, смогут эффективно адаптироваться к вызовам урбанизации, поддерживать устойчивое развитие и обеспечивать высокий уровень комфорта и безопасности для своих жителей. В дальнейшем разработка и

совершенствование данных технологий станет важным шагом на пути к созданию инновационной и устойчивой городской среды.

Список литературы

1. Архипов О.П., Иващук О.А., Константинов И.С., Савина О.А. Пути создания автоматизированной системы управления инновационным «Умным городом» //Информационные системы и технологии. 2011. №6. С. 85-94.
2. Асаул А.Н., Шуан Л. Текущие вызовы и проблемы в строительстве умных городов в Китае //Научное обозрение. Экономические науки. 2021. №2. С. 5-9.
3. Веселова А.О., Хацкелевич А.Н., Ежова Л.С. Перспективы создания «умных городов» в России: систематизация проблем и направлений их решения //Вестник Пермского университета. Серия: Экономика. 2018. Т. 13. №1. С. 75-89.
4. Коськин А.В., Архипов О.П., Иващук О.А., Пилипенко О.В., Савина О.А. Базовые принципы построения автоматизированной системы управления безопасным «умным городом» и механизмы их реализации //Строительство и реконструкция. 2012. №2. С. 63-68.
5. Королев А.С. SMART CITY: Теории и практики создания умного города //Управление городом: теория и практика. 2015. №4. С. 19-23.
6. Карпович В.Ф., Драгун К.Н. Государственно-частное партнерство как способ финансирования инфраструктурных проектов «умного» города //International journal of professional science. 2023. №4. С. 101-111.
7. Мухаметов Д.Р. Модели платформ вовлечения граждан для создания в России умных городов нового поколения //Вопросы инновационной экономики. 2020. Т. 10. №3. С. 1605-1622.
8. Давиденко Д.О., Мелентьева В.В., Татарникова М.А. От умного города к цифровому региону //Вестник Коми республиканской академии государственной службы и управления. Теория и практика управления. 2021. №2. С. 62-64.
9. Карагулян Е.А., Батырева М.В. Опыт внедрения концепции умного города в Тюменской области // Социология и общество: традиции и инновации в социальном развитии регионов. 2020. С. 3806-3813.
10. Желтышева С.Е. Проблемы на пути реализации проекта цифровизации городского хозяйства "умный город" //Вестник Белого генерала. 2020. №2. С. 24-31.
11. Федоненко М.В. Опыт развития "умных" городов в современном мире // Социально-экономические явления и процессы. 2019. Т. 14. №2(106). С. 61-72.

UDC 004.9

PROTOCOLS IN IOT: ASSESSMENT AND ENHANCEMENT

Toktosunova Z.
specialist degree, I. Razzakov Kyrgyz State Technical University (Bishkek, Kyrgyzstan)

ПРОТОКОЛЫ БЕЗОПАСНОСТИ В IOT: ОЦЕНКА И УЛУЧШЕНИЕ

Токтосунова З.А.
специалист, Кыргызский государственный технический университет им. И. Раззакова (Бишкек, Кыргызстан)

Abstract

This paper examines the main security protocols used in the Internet of Things (IoT) and analyzes methods aimed at improving data and device protection. With the growth of IoT devices, the risk of cyberattacks increases, necessitating specialized protocols like TLS, DTLS, and MQTT. TLS and DTLS provide reliable data encryption at the transport layer, though their high resource requirements limit usage in low-power devices. MQTT, optimized for low-resource devices, supports built-in authentication and encryption functions, making it popular for IoT networks. The paper also considers lightweight cryptography to enhance security with limited computational capacity and distributed access management systems based on blockchain. The combination of security protocols and adaptive methods achieves high resilience in IoT networks, enhancing overall system security and reliability.

Keywords: IoT security, TLS protocol, MQTT, DTLS, lightweight cryptography, access management.

Аннотация

В статье рассмотрены основные протоколы безопасности, используемые в интернете вещей (IoT), и проанализированы методы, направленные на улучшение защиты данных и устройств. С увеличением числа IoT-устройств возрастает риск кибератак, что требует применения специализированных протоколов, таких как TLS, DTLS и MQTT. TLS и DTLS обеспечивают надежное шифрование данных на транспортном уровне, хотя их ресурсоемкость ограничивает возможности использования в условиях ограниченных ресурсов. MQTT, оптимизированный для маломощных устройств, поддерживает встроенные функции аутентификации и шифрования, что делает его популярным для IoT-сетей. В статье также рассматривается легковесная криптография для повышения безопасности в условиях низких вычислительных мощностей, а также распределённые системы управления доступом, основанные на блокчейне. Применение комбинации протоколов безопасности и адаптивных методов позволяет достичь высокой устойчивости IoT-сетей, что повышает общую защищенность и надежность системы в целом.

Ключевые слова: безопасность IoT, протокол TLS, MQTT, DTLS, легковесная криптография, управление доступом.

Introduction

With the rise of the Internet of Things (IoT) and its integration across various sectors, including industry, healthcare, and consumer electronics, the need for securing IoT networks and devices is growing. IoT systems, which connect numerous sensors and devices into a single network, pose significant security risks, as each connected object can become a potential point of attack. Due to the

limited computational power and memory of most IoT devices, traditional security methods often prove ineffective or too resource-intensive. This article aims to examine existing security protocols for IoT and analyze methods that can enhance protection in this rapidly evolving field. One of the primary challenges for IoT protocol developers is ensuring data privacy and integrity with minimal computational resource usage. In networks composed of numerous resource-constrained devices, developing reliable protocols becomes particularly critical. In practice, this means that IoT security protocols must protect against external threats and be resilient to internal failures. This article reviews major security protocols, such as Transport Layer Security (TLS), Datagram Transport Layer Security (DTLS), and Lightweight Cryptography, their strengths and limitations, and offers recommendations for optimizing them. With the rise in cyberattacks targeting IoT devices, there is a need to analyze potential vulnerabilities and find solutions to strengthen IoT systems' resilience. This requires using flexible, scalable, and easily adaptable protocols that can integrate into various IoT networks and meet security requirements. The article examines both existing and emerging protection methods, including hybrid authentication schemes, distributed access control systems, and adaptive encryption algorithms. These approaches enhance IoT security and ensure stable network operation even as complexity increases.

Main part

One widely used protocol for securing data in IoT networks is Transport Layer Security (TLS), which encrypts data at the transport protocol level. TLS ensures data confidentiality and integrity, making it beneficial for networks with limited access. However, due to its resource-intensive nature, TLS may not be suitable for IoT devices with limited computational capabilities. In such cases, a lighter protocol version, Datagram Transport Layer Security (DTLS), is used. DTLS is specifically designed for data transmission in resource-constrained networks and supports encryption without significant device load [1].

As data may traverse numerous nodes in IoT networks, authentication becomes essential. A common approach for IoT authentication is Lightweight Cryptography, which is developed for devices with limited resources. Lightweight cryptography includes algorithms requiring minimal memory and processing power while providing a basic level of security. Examples include SPECK and Simon, which support symmetric data encryption.

Another critical aspect of IoT security is device access control [2]. In traditional networks, access control is centralized through servers, but a more suitable solution for IoT involves Distributed Access Control, allowing IoT devices to autonomously manage access without constant connection to a central server. Blockchain-based protocols are also considered a potential solution for secure access management in IoT networks, as they provide secure, immutable access records.

The Message Queuing Telemetry Transport (MQTT) protocol, which supports data protection through built-in authentication and encryption mechanisms, is used to monitor IoT device security. MQTT enables IoT devices to send and receive messages with flexibility and minimal delays [3]. An example Python code demonstrating device connection to an MQTT broker with authentication is shown below:

```
import paho.mqtt.client as mqtt

# MQTT Settings
broker = "mqtt.example.com"
port = 8883  # SSL/TLS port
username = "user"
password = "password"

# Function for SSL/TLS connection
def on_connect(client, userdata, flags, rc):
    if rc == 0:
        print("Connection successful")
    else:
```

```
        print("Connection failed, code:", rc)

# Initialize MQTT client
client = mqtt.Client()
client.username_pw_set(username, password)
client.tls_set()  # Use TLS for security
client.on_connect = on_connect

# Connect to MQTT broker
client.connect(broker, port)
client.loop_start()
```

This code demonstrates how an IoT device can connect to an MQTT broker using SSL/TLS. Authentication via MQTT ensures secure data transmission and enables the device to exchange information with other network components.

In addition to basic security, identifying anomalies and preventing attacks are crucial. Adaptive machine learning methods, which detect abnormal behavior, are used for this. These methods analyze network traffic and sensor data, identifying suspicious activity like excessive activity or unauthorized access [4]. Thus, combining security protocols with machine learning algorithms allows for a comprehensive IoT protection system, resilient to modern threats and capable of real-time data security. Figure 1 depicts the adoption trends of IoT security protocols over recent years, showing a steady increase in their implementation to secure IoT networks.

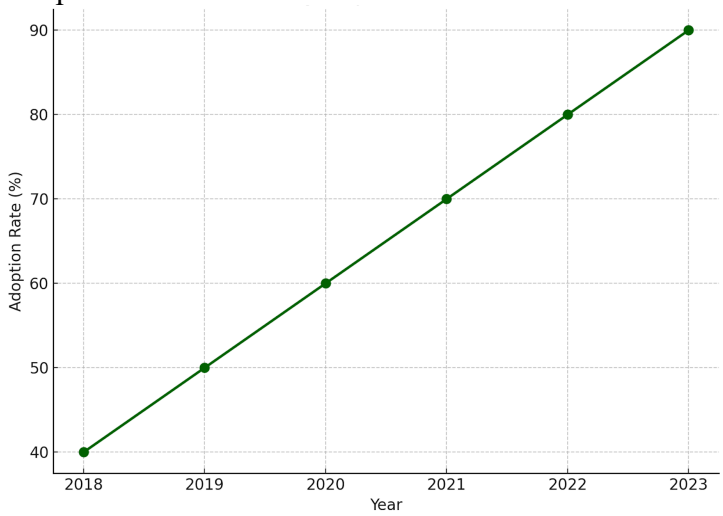

Figure 1. IoT security adoption trends over time

Figure 1 highlights a significant growth in IoT security protocol adoption, rising from 40% in 2018 to 90% in 2023. This trend reflects increasing awareness and prioritization of security measures in IoT networks. The steady increase underscores the necessity for scalable and efficient protocols to meet the growing demand for secure IoT environments.

Evaluation of IoT security protocols

Effective protection of devices and data in IoT networks requires specialized security protocols ensuring reliable data transmission and storage. IoT security protocols such as TLS (Transport Layer Security), DTLS (Datagram TLS), and MQTT (Message Queuing Telemetry Transport) play a vital role in minimizing risks related to data leaks and unauthorized access. TLS and DTLS provide secure data transmission at the network level, using encryption to protect data integrity and confidentiality. This is especially important in IoT, where devices often exchange sensitive data [5].

MQTT is a message exchange protocol widely used in IoT for transmitting data from devices to a server. Its built-in security mechanisms include client authentication and encryption support, making it ideal for situations where minimal latency and high transmission speed are required. The protocol also supports lightweight connection models, which are optimal for resource-constrained IoT devices [6].

Evaluating these protocols' effectiveness in IoT networks shows that their combined use achieves a high level of network security and resilience.

For example, TLS and DTLS provide transport-level security, particularly relevant for applications requiring reliable and secure connections. However, applying these protocols may impose significant load on devices with limited computational resources, requiring optimization [7]. Figure 2 illustrates the effectiveness of commonly used IoT security protocols, including TLS, DTLS, and MQTT, highlighting their relative performance in securing IoT networks.

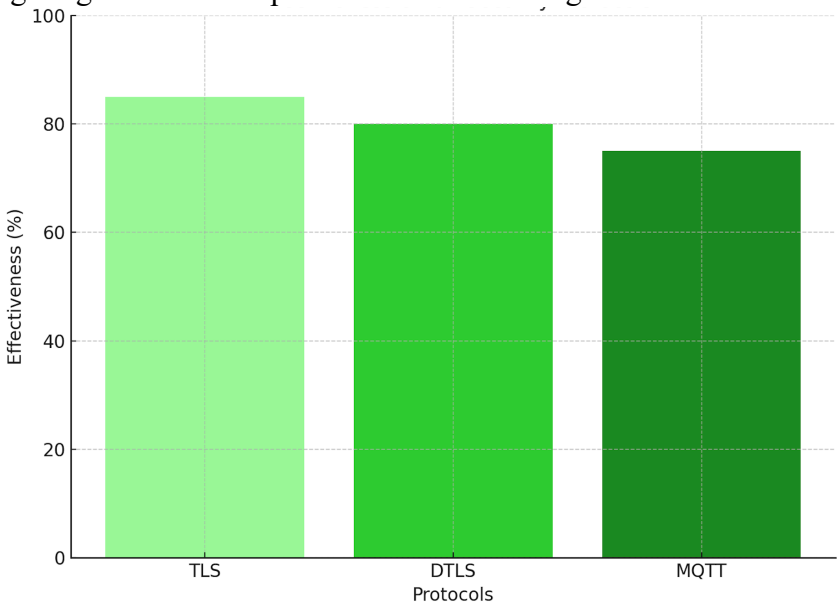

Figure 2. Effectiveness of IoT security protocols

As shown in Figure 2, TLS provides the highest level of effectiveness at 85%, ensuring robust data encryption for secure connections. DTLS follows with 80%, offering lightweight encryption suitable for resource-constrained IoT devices. MQTT, while slightly lower at 75%, demonstrates sufficient effectiveness for secure message exchanges with minimal latency. This comparison underscores the importance of selecting the appropriate protocol based on network requirements and device capabilities.

Conclusion

As the Internet of Things evolves rapidly, securing IoT networks and devices becomes a top priority. Security protocols such as TLS, DTLS, and MQTT demonstrate high potential for protecting data transmitted between devices, but their application requires consideration of IoT network specifics. TLS and DTLS, which provide transport-level security, are suitable for networks with high-security demands, though they have limitations on resource-constrained devices.

Effective implementation of IoT security protocols requires balancing protection level with device computational capacity. Using lightweight cryptographic methods and optimizing protocols like MQTT reduce device load while maintaining high security. Moreover, integrating adaptive data analysis methods and machine learning improves monitoring and allows for timely threat detection.

Thus, enhancing IoT network security requires a comprehensive approach, combining security protocols, adaptive protection methods, and distributed access control systems. These measures create robust, secure IoT networks capable of functioning effectively under high demands for data reliability and confidentiality.

References

1. Naraliev N.A., Samal D.I. Review and analysis of standards and protocols in the field of the Internet of Things. Modern testing methods and IoT information security issues // International Journal of Open Information Technologies. 2019. Vol. 7. No.8. P. 94-104.

2. Kazhenova Zh.S., Kenzhebaeva Zh.E. Security in IoT protocols and technologies: a review // International Journal of Open Information Technologies. 2022. Vol. 10. No.3. P. 10-16.

3. Umarov A., Mukhtoriddinov M., Ismoilov S. IoT security protocols: analysis, vulnerabilities, and prospects // Conference on Digital Innovation: "Modern Problems and Solutions". 2023.
4. Fomicheva V.A., Parotkin N.Yu. Review of technologies for ensuring the security of IoT device operation // Current Issues in Aviation and Cosmonautics. 2022. Vol. 2. P. 318-320.
5. Baev D.A., Volkov R.O., Zonov A.D. Security monitoring in IoT networks // StudNet. 2021. Vol. 4. No.6. P. 1119-1126.
6. Ermakov S.A., Bolgov A.A., Mokrousov A.N. Assessment of IoT network protection efficiency using the example of smart home technology // Information and Security. 2019. Vol. 22. No.1. P. 130-133.
7. Hofer-Schmitz K., Stojanović B. Towards formal verification of IoT protocols: A Review // Computer Networks. 2020. Vol. 174. P. 107233.