

ПРИМЕНЕНИЕ МЕТОДОВ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА ДЛЯ ОПТИМИЗАЦИИ АЛГОРИТМОВ СЖАТИЯ ДАННЫХ

И.В. ВЕРГЕЛЬ, С.Н. БАРСУКЕВИЧ, С.А. МИГАЛЕВИЧ

Белорусский государственный университет информатики и радиоэлектроники, Республика Беларусь

Поступила в редакцию 20 марта 2025

Аннотация. Рассматриваются современные подходы к сжатию данных, включая традиционные алгоритмы и методы, основанные на искусственном интеллекте. Анализируются преимущества и недостатки различных технологий, их вычислительная сложность, степень сжатия и применимость к разным типам данных. Внимание уделяется нейросетевым методам, таким как сверточные нейронные сети, генеративные состязательные сети и языковые модели. Приведено сравнение эффективности различных алгоритмов на основе тестирования сжатия текстовых данных.

Ключевые слова: алгоритмы сжатия данных, искусственный интеллект, нейронные сети

Введение

Сжатие данных является одной из ключевых технологий в современной информатике, позволяющей эффективно хранить, передавать и обрабатывать большие массивы данных. С развитием цифровых технологий объемы данных, генерируемых ежедневно, достигли невероятных масштабов. По оценкам экспертов, в 2025 году мировой объем данных может составить 175 зеттабайт (1 зеттабайт = 10^{21} байт), что создает огромную нагрузку на системы хранения и передачи информации. В таких условиях сжатие данных становится не просто полезным инструментом, а необходимостью для обеспечения устойчивости цифровой инфраструктуры. Традиционные методы сжатия достигли значительных успехов, однако с ростом объемов данных возникла необходимость в более совершенных подходах.

Современные подходы к сжатию данных

Сжатие данных – это ключевая технология, которая позволяет эффективно хранить и передавать информацию. Эта технология делится на два ключевых направления: восстановление данных в исходном виде и обработка с частичным удалением информации. При сжатии без потерь данные восстанавливаются в исходном виде после распаковки, что особенно важно для текстов, программного кода и медицинских изображений. Сжатие с потерями, напротив, жертвует частью данных, но это обычно остается незаметным для пользователя, особенно в мультимедийных файлах, таких как изображения, аудио и видео.

Традиционные методы сжатия, такие как алгоритмы Хаффмана, LZ77 или JPEG, основаны на математических преобразованиях и статистических моделях. Они эффективны, но часто не учитывают семантику данных, что ограничивает их потенциал. Кроме того, многие из этих алгоритмов работают медленно из-за итеративного подхода, который не позволяет эффективно использовать параллельные вычисления.

Искусственный интеллект (ИИ) предлагает революционные подходы к сжатию данных, которые превосходят традиционные методы по степени сжатия и качеству восстановления. Нейронные сети, особенно сверточные нейронные сети (CNN), активно применяются для сжатия изображений. Они анализируют структуру изображения, выделяют важные детали и удаляют избыточную информацию с минимальными потерями качества. Основным принципом работы CNN

в сжатии изображений заключается в том, что сеть проходит через слои свертки и пуллинга, выделяя значимые признаки и понижая разрешение входных данных, а затем использует декодирующую часть для восстановления изображения. Например, алгоритмы на основе ИИ могут достигать сжатия изображений в 10 раз эффективнее, чем JPEG, при сохранении визуально приемлемого качества.

Генеративные модели, такие как Generative Adversarial Networks (GAN), используются для восстановления данных после сжатия с потерями. Генератор создает данные, а дискриминатор оценивает их качество, позволяя восстанавливать данные с минимальными потерями. GAN работают на основе конкурентного обучения: генератор пытается создать изображения, похожие на исходные, а дискриминатор учится отличать реальные изображения от сгенерированных. Этот процесс приводит к улучшению качества восстановленных данных и делает GAN незаменимыми в задачах, где важна визуальная точность.

Для текстового сжатия нейронные языковые модели, такие как GPT и BERT, предсказывают следующие слова и удаляют избыточные данные, что позволяет достигать высокой степени сжатия без потери смысла. Эти модели работают на основе трансформерной архитектуры, где механизм внимания позволяет анализировать длинные зависимости в тексте и учитывать контекст при кодировании и декодировании информации. Такие алгоритмы способны адаптироваться под специфику текста, будь то научная статья, художественная литература или техническая документация.

В таблице 1 приведены ключевые отличия традиционных и ИИ-алгоритмов.

Табл. 1. Ключевые отличия традиционных алгоритмов и ИИ-алгоритмов

Аспект	Традиционные алгоритмы	ИИ-алгоритмы
Основа метода	Статистические модели или математические преобразования.	Нейросетевые модели, обученные на данных.
Адаптивность	Фиксированные правила, не адаптируются к специфике данных.	Могут адаптироваться под конкретные типы данных.
Степень сжатия	Ограничена predetermined шаблонами	Выше за счет семантики и сложных закономерностей.
Вычислительная сложность	Низкая.	Высокая.
Восстановление данных	Точное восстановление только при сжатии без потерь.	Возможность восстановления утраченных деталей.

Одним из ключевых преимуществ ИИ-алгоритмов является их способность находить сложные закономерности в данных и оптимизировать их хранение. Например, нейронные сети могут анализировать контекст изображения или текста, что позволяет им более эффективно сжимать информацию.

Однако у ИИ-методов есть и свои недостатки. Для их обучения и работы необходимы мощные графические процессоры (GPU) или тензорные процессоры (TPU), которые способны выполнять параллельные вычисления с высокой скоростью. Это особенно важно для задач, связанных с обработкой изображений, видео и текстов, где объемы данных могут быть чрезвычайно большими. Для эффективного использования ИИ-алгоритмов в реальном времени требуется оптимизация их работы на устройствах с ограниченными ресурсами, таких как мобильные устройства или встраиваемые системы.

Сложность алгоритмов делает их менее прозрачными по сравнению с традиционными методами. Нейронные сети, используемые для сжатия, могут непреднамеренно сохранять или восстанавливать конфиденциальную информацию, что создает риски утечки данных. Например, генеративные модели, такие как GAN, могут восстанавливать детали изображений, которые были намеренно удалены. Это требует разработки новых методов защиты данных, таких как дифференциальная приватность и шифрование на этапе сжатия. Это может вызывать вопросы в областях, где важна интерпретируемость, например, в медицине или юриспруденции.

Также недостатком ИИ-алгоритмов является их чрезвычайно высокая энергозатратность. Обучение сложных нейронных сетей требует огромных вычислительных ресурсов, что приводит к потреблению сотен мегаватт-часов электроэнергии – сопоставимо с энергопотреблением небольшого города. Эта проблема усугубляется тенденцией к увеличению масштабов моделей, где число параметров достигает сотен миллиардов, что влечет за собой экспоненциальный рост энергопотребления и выбросов углекислого газа.

Сравнение эффективности алгоритмов сжатия текстовых данных

Для анализа эффективности алгоритмов сжатия текстовых данных обратимся к результатам тестирования, сравнив ключевые параметры: степень сжатия, скорость работы и вычислительную сложность. Это позволит понять, насколько традиционные алгоритмы уступают современным подходам на основе ИИ.

Традиционные алгоритмы сжатия, такие как GZIP, BZIP2 и LZMA, остаются популярными благодаря своей простоте, высокой скорости и низкой вычислительной сложности. Однако они значительно уступают ИИ-подходам в степени сжатия и адаптивности. Например, алгоритм GZIP, основанный на методе DEFLATE, сжимает данные до ~35% от исходного размера, что является хорошим результатом для повседневных задач, но недостаточным для работы с большими объемами данных.

Современные ИИ-алгоритмы, такие как LSTM (Long Short-Term Memory) и CM (Context Mixing), демонстрируют более высокую степень сжатия, достигая ~12% и ~11% от исходного размера соответственно. Это стало возможным благодаря использованию нейронных сетей, которые анализируют семантику текста и находят сложные закономерности, недоступные для традиционных методов.

В таблице 2 приведены результаты сравнения эффективности различных алгоритмов.

Табл. 2. Сравнение эффективности алгоритмов

Алгоритм	Степень сжатия	Скорость сжатия	Вычислительная сложность
PAQ8	~15% от исходного размера	Очень медленная	Очень высокая
GZIP	~35% от исходного размера	Быстрая	Низкая
BZIP2	~25% от исходного размера	Средняя	Средняя
LZMA	~20% от исходного размера	Медленная	Высокая
ZSTD	~30% от исходного размера	Очень быстрая	Низкая
PPMd	~18% от исходного размера	Медленная	Высокая
Tr	~15% от исходного размера	Быстрая	Высокая
CM	~11% от исходного размера	Быстрая	Высокая
LSTM	~12% от исходного размера	Быстрая	Высокая

Традиционные алгоритмы сжатия остаются актуальными благодаря своей простоте и скорости, но значительно уступают ИИ-подходам в степени сжатия и адаптивности. Например, GZIP, ZSTD и LZ4 идеально подходят для задач, где важна скорость обработки, таких как передача данных в реальном времени, сжатие логов или потоковое сжатие мультимедийных данных. Эти алгоритмы используют метод кодирования LZ77 или его вариации, обеспечивая хорошее сжатие при высокой скорости декомпрессии.

Однако для задач, требующих максимальной степени сжатия, например, архивирования больших текстовых баз данных, более эффективны ИИ-алгоритмы, такие как LSTM, Transformer-based модели и контекстные смещающие алгоритмы (CM). Эти методы используют нейросетевые подходы для моделирования вероятностей появления символов или слов, что позволяет добиться более высокой степени компрессии по сравнению с традиционными методами. Тем не менее, высокая вычислительная сложность ИИ-алгоритмов и требовательность к данным и ресурсам ограничивают их применение на маломощных устройствах и в реальном времени.

Одним из ключевых преимуществ ИИ-алгоритмов является их способность адаптироваться к специфике данных. Например, LSTM может эффективно сжимать тексты на естественных языках, учитывая их грамматическую структуру и семантику, что недоступно для традиционных методов. Аналогично, алгоритмы на основе трансформеров могут строить сложные языковые модели, анализируя контекст слов и предложений, что позволяет уменьшать избыточность данных.

Кроме того, эффективность алгоритмов может варьироваться в зависимости от типа текстовых данных. Например, для сжатия технической документации с высокой избыточностью лучше подходят традиционные методы, такие как LZMA или BZIP2, поскольку они эффективно обрабатывают повторяющиеся последовательности символов. В то время как для художественных текстов с богатой семантикой более эффективны ИИ-алгоритмы, способные учитывать сложные языковые структуры и предсказывать вероятностное распределение слов.

Стоит также отметить развитие гибридных методов сжатия, комбинирующих традиционные алгоритмы и нейросетевые подходы. Например, алгоритмы типа PAQ, которые используют контекстное смешивание и статистическое моделирование, демонстрируют высокую степень сжатия, приближенную к ИИ-методам, при этом оставаясь более доступными

по вычислительным требованиям. Подобные гибридные модели могут найти широкое применение в системах хранения данных, облачных сервисах и распределенных вычислениях.

Заключение

ИИ-алгоритмы сжатия данных предлагают значительные преимущества по сравнению с традиционными методами, включая более высокую степень сжатия и адаптивность к различным типам данных. Однако их применение требует значительных вычислительных ресурсов, что создает проблемы с энергопотреблением и экологией. Для выбора подходящего алгоритма сжатия необходимо учитывать тип данных, требования к скорости, степени сжатия и качеству восстановления. Для текстовых данных с высокими требованиями к степени сжатия подходит LSTM или трансформеры. Для изображений и мультимедиа: CNN, GAN. Для задач с ограниченными вычислительными ресурсами: GZIP, ZSTD. Дальнейшее развитие ИИ-алгоритмов сжатия данных будет связано с оптимизацией их энергопотребления, повышением безопасности и разработкой новых стандартов для обработки больших объемов информации.

APPLICATION OF ARTIFICIAL INTELLIGENCE METHODS TO OPTIMIZE DATA COMPRESSION ALGORITHMS

I.V. VERGHEL S.N. BARSUKEVICH S.A. MIGALEVICH

Abstract. Modern approaches to data compression, including traditional algorithms and methods based on artificial intelligence, are considered. The advantages and disadvantages of various technologies, their computational complexity, compression ratio, and applicability to different types of data are analyzed. Attention is paid to neural network methods such as convolutional neural networks, generative adversarial networks, and language models. The effectiveness of various algorithms based on text data compression testing is compared.

Keywords: data compression algorithms, artificial intelligence, neural networks

Список литературы

1. Qi Wu, David Bauer, Michael J. Doyle, Kwan-Liu Ma // Instant Neural Representation for Interactive Volume Rendering. 2022. Режим доступа: <https://arxiv.org/abs/2207.11620v1>. Дата доступа: 11.02.2025.
2. В. В. Кудинов, О. К. Сапрыкин // Вопросы электромеханики. 2024. Том 201 №4 с. 37-41. Режим доступа: <https://jurnal.vniem.ru/text/201/37-41.pdf>. Дата доступа: 11.02.2025.
3. М.А. Шарапов., А.В. Левенец // Информационные Технологии XXI Века. 2023. с 71-74. Режим доступа: https://www.elibrary.ru/download/elibrary_53737268_90436074.pdf. Дата доступа: 10.02.2025.
4. Large Text Compression Benchmark [Электронный ресурс]. Режим доступа: <http://matthmahoney.net/dc/text.html>. Дата доступа: 11.02.2025.
5. Neha Sharma, Usha Batra // EAI Endorsed Transactions on Scalable Information Systems. 2020. Режим доступа: https://www.researchgate.net/publication/339742479_Performance_analysis_of_compression_algorithms_for_information_security_A_Review. Дата доступа: 10.02.2025.