

УДК 004.8

Архитектура автоэнкодера на основе трансформера: низкоуровневое представление 3D-объектов

Кочнов Андрей Павлович

Волосатова Тамара Михайловна (научный руководитель)

МГТУ им. Н.Э. Баумана, Москва, Россия

Рассмотрена архитектура автоэнкодера на основе трансформера, предназначенная для обработки и оптимизации 3D-объектов. Модель включает несколько ключевых компонентов: дискретизацию координат, графовые сверточные сети для захвата локальных геометрических особенностей, механизмы внимания для оптимизации вычислений, а также остаточное квантование, которое эффективно сжимает данные, сохраняя их точность. Такая архитектура позволяет компактно представлять 3D-объекты на низком уровне, облегчая генерацию и восстановление. Практическое применение данного автоэнкодера в САПР позволяет компактно и точно представлять 3D-объекты на низком уровне, облегчая их генерацию и восстановление. Модель позволит улучшить существующие CAD-системы, а также может стать отправной точкой в развитии автоматических систем проектирования, такие как text-to-design системы.

Ключевые слова: искусственный интеллект, автоэнкодер, токенизация, остаточное квантование, трансформер

Введение. Современные системы искусственного интеллекта, такие как большие языковые модели, показывают отличные результаты в плане генерации осмысленного текста, а также создания картинок. Большинство таких моделей являются авторегрессионным, что позволяет им эффективно и качественно создавать требуемый контент. В отличие от них, область создания 3D-объектов, в основном использует технологию диффузии [1], что, в свою очередь, проблемно интегрировать с системы автоматизированного проектирования (САПР), из-за невозможности работы с конкретными деталями объекта. Автоэнкодер, основанный на трансформере и способный эффективно кодировать 3D-объекты в токены, может быть интегрирован в большие языковые модели, создавая таким образом модели генеративного проектирования, с полным контролем над деталями объекта.

Архитектура: входные данные и токенизация. На первом этапе модель получает на вход набор вершин и граней, где вершины представляют собой координаты в трехмерном пространстве. Для граней вычисляются их площади, углы и нормали. Далее координаты вершин проходят процесс дискретизации и эмбединга. Площади, углы и нормали граней проходят этот процесс. Все данные конкатенируются и подаются дальше.

Архитектура: строение энкодера. Энкодер модели сочетает в себе графовые сверточные сети и механизмы внимания. Графовые сверточные сети

[2] обеспечивают распространение информации между соседними гранями сетки, что позволяет фиксировать локальные геометрические особенности. Эта операция описывается уравнением

$$h_i^{(l+1)} = W^{(l)} \cdot MEAN \left(\{h_i^{(l)}, h_j^{(l)} : j \in N(i)\} \right) + b^{(i)}, \quad (1)$$

где $h_i^{(l)}$ — вектор признаков вершины i на l -м слое; $h_i^{(l+1)}$ — новый вектор признаков вершины i на $(l+1)$ -м слое; $N(i)$ — множество соседей вершины i на графе; $MEAN(\{\cdot\})$ — операция усреднения по всем вхождениям; $W^{(l)}$ — обучаемая матрица весов на l -м слое; $b^{(i)}$ — обучаемый вектор смещения; $h_j^{(l)}$ — вектор признаков для вершины-соседа j , также на l -м слое.

В основе слоя внимания лежат два типа внимания: локальное многоголовое внимание и линейное внимание на основе ряда Тейлора. Локальное внимание ограничивает взаимодействие токенов соседними, снижая сложность. Линейное внимание использует аппроксимацию ряда Тейлора, что также уменьшает вычислительную сложность [3, 4], сокращая количество операций по сравнению с классическим softmax-вниманием.

Архитектура: остаточное квантование. Один из ключевых компонентов модели — это остаточное квантование [5], которое используется для сжатия эмбедингов сетки. Вместо того, чтобы квантовать сетку напрямую, оно последовательно квантует остатки информации, что позволяет более точно захватывать мелкие детали сетки:

$$q_i^{(D+1)} = C^{(D+1)} \left(\arg \min_{c \in C^{(D+1)}} \|r_i^{(D)} - c\|^2 \right), \quad (2)$$

где $r_i^{(D)}$ — остаток после D этапов квантования для объекта i ; $C^{(D+1)}$ — кодбук на этапе $D+1$; c — конкретный код из кодбука; $q_i^{(D+1)}$ — выбранный эмбединг из кодбука, ближайший к остатку.

Далее получается формула обновления остатка:

$$r_i^{(D+1)} = r_i^{(D)} - q_i^{(D+1)}. \quad (3)$$

Окончательное квантованный эмбединг представляет собой сумму всех квантованных компонентов с каждого этапа:

$$q_i = \sum_{d=1}^D q_i^{(d)}.$$

Каждый эмбединг грани представляется стеком D кодов, где каждый код соответствует квантованному остатку от предыдущего этапа. Кодбук C обучается с использованием остаточного векторного квантования, которое гарантирует, что кодбук каждого этапа адаптируется к остаткам предыдущего этапа. Формально набор эмбединга для грани i после D этапов квантования можно записать как

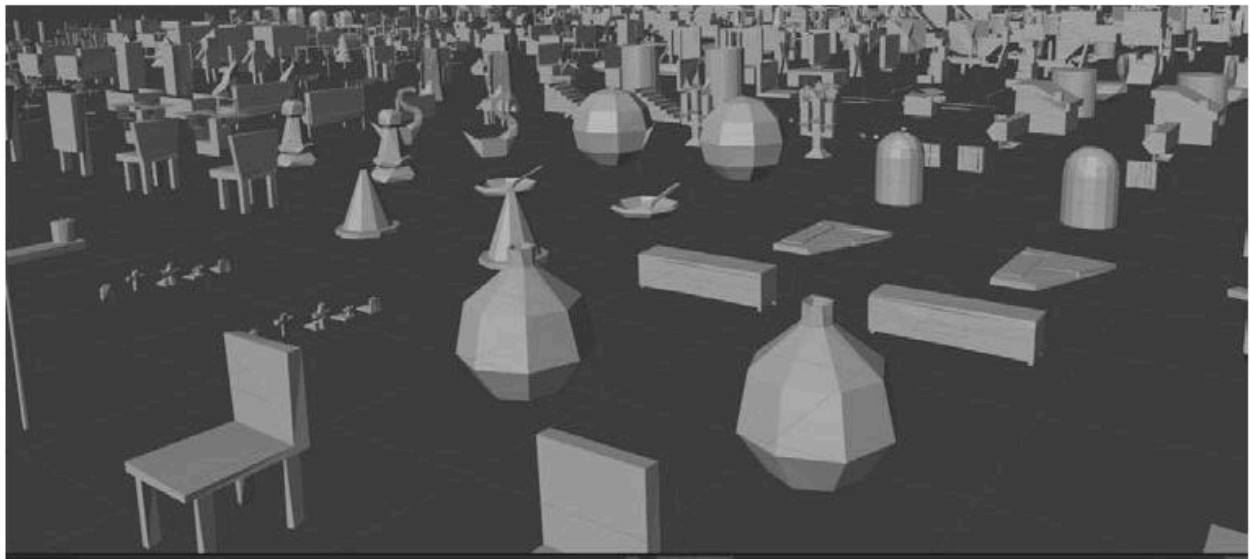
$$T_i = (t_i^1, t_i^2, \dots, t_i^{(D)}), t_i^{(d)} \in \mathcal{C}^{(D)}, \quad (4)$$

где T_i — набор, представленный из индексов кодов, выбранных на каждом этапе квантования;

Этот набор токенов затем используется в качестве входных данных для декодера для реконструкции сетки.

Архитектура: декодер и реконструкция. Заключительный этап автоэнкодера включает декодирование квантованных эмбедингов обратно в координаты 3D-сетки. Квантованные токены проходят через блоки внимания и сверточные слои ResNet для преобразования в вектор признаков. Финальная сетка формируется из трех вершин каждой грани, создавая компактную и точную модель, сохраняющую все геометрические детали.

Обучение и результаты. Для обучения автоэнкодера был собран датасет из 50 тысяч 3D-моделей, которые имеют не более 250 граней. Модели были собраны из таких датасетов как: ShapeNet и Objaverse. Тренировка происходила на двух видеокартах Nvidia V100 в течение двух дней. Для небольшого количества моделей с малым числом граней, был выбран размер кодбука в 2048 кодов. В результате обучения модель научилась эффективно представлять 3D-объекты в виде набора кодов, с последующим их восстановлением. На рисунке продемонстрировано сравнение оригинальной модели и модели, которая была закодирована и восстановлена.



Сравнение оригинальной модели и закодировано-восстановленной

Заключение. Представленная модель автоэнкодера с архитектурой на основе трансформеров демонстрирует высокую эффективность в задаче кодирования и восстановления 3D-моделей, благодаря чему способна преобразовывать сложные их в компактное дискретное представление, пригодное для дальнейшей генерации. Интеграция этого токенизатора с архитектурой боль-

шой языковой модели, такой как GPT, открывает перспективы для построения трансформеров нового поколения, способных порождать 3D-объекты по текстовому описанию. Такой подход позволит реализовать интуитивное моделирование объектов с использованием естественного языка, что способно улучшить текущие CAD-системы, а также дать развитие text-to-design системам и системам генеративного проектирования.

Однако текущая реализация имеет ряд ограничений. Одним из ключевых является ограниченная способность модели представлять и восстанавливать гладкие и криволинейные поверхности. Это связано с дискретной природой токенизации и недостаточной выразительностью эмбеддингов в локальных геометрически сложных областях. Решение данной проблемы требует дальнейших исследований, в том числе в направлении более гибких методов представления поверхности.

Литература

- [1] Peebles W., Xie S. Scalable Diffusion Models with Transformers. Arxiv:2212.09748, 2022. <https://doi.org/10.48550/arXiv.2212.09748>
- [2] Hamilton W.L., Ying R., Leskovec J. Inductive Representation Learning on Large Graphs. Arxiv:1706.02216, 2017. <https://doi.org/10.48550/arXiv.1706.02216>
- [3] Keles F.D., Wijewardena P.M., Hegde Ch. On The Computational Complexity of Self-Attention. Arxiv:2209.04881, 2022. <https://doi.org/10.48550/arXiv.2209.04881>
- [4] Simran A., Sabri E., Timalisina A., Johnson I., Poli M., Zou J., Rudra A., Ré Ch. Zoology: Measuring and Improving Recall in Efficient Language Models. Arxiv:2312.04927, 2023. <https://doi.org/10.48550/arXiv.2312.04927>
- [5] Zeghidour N., Luebs A., Omran A., Skoglund J., Tagliasacchi M. SoundStream: An End-to-End Neural Audio Codec. Arxiv:2107.03312, 2021. <https://doi.org/10.48550/arXiv.2107.03312>

Transformer-based autoencoder architecture: low-level representation of 3D objects

Kochnov Andrey Pavlovich

Volosatova Tamara Mikhailovna (*scientific advisor*)

BMSTU, Moscow, Russia

This research presents a transformer-based autoencoder architecture for processing and optimizing 3D objects. It includes coordinate discretization, graph convolutional networks for capturing local geometry, attention mechanisms for efficient computation, and residual quantization for compact yet accurate data representation. The model enables low-level, compact encoding of 3D objects, supporting generation and reconstruction. Its application in CAD systems can enhance precision and efficiency, serving as a foundation for future automated design systems, including text-to-design solutions.

Keywords: artificial intelligence, autoencoder, tokenization, residual quantization, transformer