УДК 004.8

Диффузионные модели для синтеза исходного кода: архитектурные парадигмы и перспективы применения

Ро А.С., Старцев Д.С. (МИЭМ НИУ ВШЭ)
Научный руководитель – кандидат технических наук, профессор Авдошин С.М. (МИЭМ НИУ ВШЭ)

Введение. Автоматический синтез исходного кода является одной из ключевых задач современной программной инженерии. Несмотря на успехи авторегрессионных трансформерных моделей (Codex, CodeT5, StarCoder), их последовательный характер генерации (слева направо) ограничивает возможности глобального редактирования и исправления ошибок. Диффузионные модели предлагают альтернативный итеративный механизм генерации, основанный на восстановлении данных из зашумлённого состояния [8]. Такой подход обеспечивает двусторонний контекст на каждом шаге деноизинга, что особенно ценно для структурированных данных, каким является код. В зарубежных и отечественных исследованиях всё большее распространение получают методы, адаптирующие диффузию для работы с программным кодом [1-6], однако область находится на этапе активного формирования, а сравнительный анализ подходов отсутствует. В связи с этим актуальной является задача систематизации современных архитектур и методов оценки диффузионных моделей для синтеза кода.

Основная часть. Обзор основан на анализе публикаций из IEEE Xplore, ACM Digital Library, SpringerLink и arXiv за 2022–2026 гг. Из 53 работ отобрано 10, посвящённых применению диффузионных моделей к синтезу исполняемого кода с экспериментальной оценкой. Рост числа публикаций в 2024–2025 гг. показывает активное формирование направления. Диффузионные архитектуры для синтеза кода делятся на три типа:

Латентные - выполняют диффузию в скрытом непрерывном пространстве эмбеддингов (например, CODEFUSION [1]). Обеспечивают плавное представление, но могут терять синтаксическую и семантическую целостность.

Дискретные маскированные - работают с токенами кода, используя маскирование и восстановление последовательности (Mercury [7], Dream 7B, LLaDA 1.5). Обеспечивают хорошие результаты в инфиллинге, но склонны к ошибкам без структурных ограничений.

Структура-осознанные - учитывают синтаксис и семантику с помощью AST-маскировки [4, 5], CFG-ограничений и компиляторного ремаскирования (CodeDiffuSe).

Наиболее распространённым подходом остаётся дискретная маскированная диффузия, а структура-осознанные и латентные методы развиваются как специализированные ветви.

По сравнению с авторегрессивными LLM [3], диффузионные модели демонстрируют: 1)сопоставимую функциональную корректность (Pass@k), 2) повышенную синтаксическую валидность (до 92% AST-validity [4,5,6] у CodeDiffuSe), 3) улучшенные результаты в инфиллинге, исправлении ошибок и генерации по спецификации благодаря итеративной природе обучения.

Существенным ограничением диффузионных моделей является высокая вычислительная сложность многошагового деноизинга. Для её снижения предлагаются ускоренные сэмплеры, в частности FastdLLM, а также методы адаптивного сокращения шагов и параллельного декодирования [9, 10]. Дополнительно развиваются гибридные LLM+диффузионные архитектуры, обеспечивающие компромисс между точностью генерации и вычислительной эффективностью. Подход сохраняет высокий потенциал для задач синтеза, исправления и рефакторинга программного кода

Выводы. Диффузионные модели для синтеза кода формируют новую парадигму, в рамках которой выделены три архитектурных класса; дискретные маскированные и структура-осознанные подходы демонстрируют высокую точность в задачах инфиллинга и исправления ошибок и особенно перспективны как «интеллектуальный редактор» поверх быстрых авторегрессионных LLM в гибридных системах генерации кода. Их неавторегрессивная природа с двусторонним контекстом и итеративным уточнением делает такие модели особенно подходящими для задач, где важны глобальная согласованность и структурная целостность программ. Ключевые направления продолжения работы связаны с ускорением многошагового деноизинга (специализированные сэмплеры, KV cache, адаптивные схемы) и повышением интерпретируемости за счёт методов объяснимого ИИ, а также переходом к масштабным экспериментам и дообучению моделей на задачах инфиллинга, исправления ошибок и генерации по спецификации на суперкомпьютерном комплексе НИУ ВШЭ «cHARISMa».

Благодарности. Исследование выполнено с использованием суперкомпьютерного комплекса НИУ ВШЭ.

Список использованных источников:

1. Singh M., Cambronero J., Gulwani S., et al. CODEFUSION: A Pre-trained Diffusion Model for Code Generation. Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing (EMNLP), 2023, pp. 11697–11708. URL: https://arxiv.org/abs/2310.17680 (дата обращения: 21.02.2026).
2. Gong S., Zhang R., Zheng H., et al. DiffuCoder: Understanding and Improving Masked Diffusion Models for Code Generation. arXiv, 2025. URL: https://arxiv.org/abs/2506.20639 (дата обращения: 21.02.2026).
3. Li C., Zhang Y., Li J., et al. Beyond Autoregression: An Empirical Study of Diffusion Large Language Models for Code Generation. arXiv, 2025. URL: https://arxiv.org/abs/2509.11252 (дата обращения: 21.02.2026).
4. Xu G., Shi Q., Zhang D., et al. Diffusion On Syntax Trees For Program Synthesis. arXiv, 2024. URL: https://arxiv.org/abs/2405.20519 (дата обращения: 21.02.2026).
5. Zeng Y., Cao J., Li Z., et al. TreeDiff: AST-Guided Code Generation with Diffusion LLMs. arXiv, 2025. URL: https://arxiv.org/abs/2508.01473 (дата обращения: 21.02.2026).
6. Anonymous. Diffusion Is a Code Repair Operator and Generator. arXiv, 2025. URL: https://arxiv.org/abs/2502.01384 (дата обращения: 21.02.2026).
7. Khanna S., Kharbanda S., Li S., et al. Mercury: Ultra-Fast Language Models Based on Diffusion. arXiv, 2025. URL: https://arxiv.org/abs/2506.17298 (дата обращения: 21.02.2026).
8. Li T., Chen M., Guo B., et al. A Survey on Diffusion Language Models. arXiv, 2025. URL: https://arxiv.org/abs/2508.10875 (дата обращения: 21.02.2026).
9. Christopher J. K., Bartoldson B. R., Ben-Nun T., et al. Accelerating Diffusion LLMs via Adaptive Parallel Decoding. arXiv, 2025. URL: https://arxiv.org/abs/2506.10848 (дата обращения: 21.02.2026).
10. Israel D., Van den Broeck G., Grover A. FastdLLM: Training-Free Acceleration of Diffusion LLM by Enabling KV Cache and Parallel Decoding. arXiv, 2025. URL: https://arxiv.org/abs/2505.21467 (дата обращения: 21.02.2026).