УДК 004.8:004.42

ИССЛЕДОВАНИЕ ВЛИЯНИЯ ТЕМПЕРАТУРЫ СЭМПЛИРОВАНИЯ НА КАЧЕСТВО ГЕНЕРАЦИИ ПРОГРАММНОГО КОДА БОЛЬШОЙ ЯЗЫКОВОЙ МОДЕЛЬЮ НА МУЛЬТИЯЗЫЧНЫХ ДАННЫХ

С.В. Гладков[1], И.А. Килбас[1], Р.А. Паринге[1]

[1]Самарский университет, г. Самара
e-mail: gerrhellpony@gmail.com

Аннотация

В работе исследуется влияние температуры сэмплирования и алгоритма top-k на качество генерации программного кода моделью Qwen3.5-4B-Base на англоязычном наборе HumanEval и его русском варианте из mHumanEval. Качество оценивается с помощью метрик pass@k для задач на Python. Дополнительно анализируются логические и синтаксические ошибки. Показано, что англоязычные постановки стабильно обеспечивают более высокие результаты, качество существенно зависит от температуры, а дополнительная настройка с алгоритмом top-k даёт дальнейший прирост в метрике.

Ключевые слова: большие языковые модели, сэмплирование, температура, генерация кода, HumanEval, mHumanEval, Qwen 3.5.

Введение

Большие языковые модели активно применяются для генерации программного кода, а их качество обычно оценивается по функциональной корректности [1] на англоязычных бенчмарках, прежде всего HumanEval [2]. Однако при переходе к неанглоязычным постановкам задач результаты могут заметно изменяться, что делает актуальным использование мультиязычных наборов данных [3], в частности mHumanEval. Для задач генерации кода важен и выбор параметров декодирования [4]. Температура управляет балансом между детерминированностью и разнообразием, а алгоритм декодирования top-k ограничивает некоторое множество кандидатов на каждом шаге генерации. Поэтому практический интерес представляет совместный анализ влияния языка постановки задачи и параметров декодирования на качество сгенерированного кода. В данной работе исследуется влияние температуры сэмплирования и алгоритма top-k на качество генерации Python-кода моделью Qwen3.5-4B-Base на наборах HumanEval и mHumanEval.

1. Методология

Для проведения исследования используется локальная предобученная большая языковая модель Qwen3.5-4B-Base [5].

Для сэмплирования основным параметром была выбрана температура в классическом диапазоне от *0,2* до *1,0* с шагом *0,2*. Он позволяет рассматривать основные режимы генерации моделью, как детерминированные, так и более стохастические, позволяя модели экспериментировать в рамках генераций. Также применяется алгоритм декодирования *top-k* для подбора значения *k*, улучшающего качество генерации.

Для оценки качества программного кода был выбран бенчмарк HumanEval, в паре с мультиязычной версией – mHumanEval [3], для сравнительного анализа результатов эксперимента на двух языках. Набор данных включает в себя 164 задачи общего программирования, конкретно выбранный набор рассматривает только язык Python.

Метрика оценки качества программного кода была выбрана *pass@k* [2], при которой генерируется *k* решений задачи, задача считается решенной, если хотя бы одна из генераций прошла все тесты. В самом эксперименте рассматривается одиночная генерация (pass@1) и *pass@5*. В качестве основного критерия выбора лучшей температуры использовалось максимальное значение *pass@1*; при близких значениях дополнительно учитывалась *pass@5*. Метрика *pass@1* рассматривалась как основная для сценария одиночной генерации, а *pass@5* – как дополнительная характеристика устойчивости качества при множественной генерации. Далее алгоритм декодирования top-k применялся отдельно при фиксированной лучшей температуре для каждого языка.

2. Постановка эксперимента

Исследование сэмплирования для каждого из значений температур дает решения всех задач из набора с 5 генерациями на задачу, для одиночной – с 1 генерацией.

Также в ходе эксперимента получены данные об ошибках в исполняемом коде, которые возвращает среда исполнения от интерпретатора языка Python, сама среда выполнения использует python 3.10-slim внутри контейнера Docker.

3. Результаты

Результаты эксперимента, к каждому значению из диапазона и метрики, представлены в таблице 1 и 2 для английского и русского языков соответственно.

Таблица 1 – Результаты для английского языка, полученные на наборе HumanEval

Температура	**Pass@1, %**	**Pass@5, %**	**Логика**	**Синтаксис**	**Кол-во ошибок**
0,2	71,3	77,4	15	217	232
0,4	78,7	77,2	19	203	222
0,6	**82,3**	**77,7**	31	181	**212**
0,8	75,6	74,9	75	189	246
1,0	60,4	63,3	118	248	366

Таблица 2 – Результаты для русского языка, полученные на наборе mHumanEval

Температура	**Pass@1, %**	**Pass@5, %**	**Логика**	**Синтаксис**	**Кол-во ошибок**
0,2	56,1	57,2	35	388	423
0,4	66,5	64,5	66	298	346
0,6	72,0	67,3	83	231	314
0,8	**72,6**	**67,6**	126	185	**311**
1,0	41,5	47,4	207	320	527

В результате были выбраны и зафиксированы наилучшие значения температуры для обоих языков и проведен дополнительный эксперимент по поиску оптимального значения параметра *k* для алгоритма *top-k* с целью дальнейшего улучшения процесса сэмплирования, результаты представлены в таблице 3.

Таблица 3 – Результаты обоих языков, полученные для параметра *k*

Английский набор	Pass@1, %	Top-k	Русский набор	Pass@1, %	Top-k
	73,1	1		58,9	1
	83,5	**10**		**73,3**	**10**
	76,6	50		65,0	50
	77,8	100		66,4	100

Из результатов видно, что англоязычный набор задач HumanEval стабильно показывает более высокое качество, чем русскоязычные переводы из mHumanEval. Так, при температуре *0,6* для английского набора значение *pass@1* составило 82,3% против 72,0% для русского набора, а *pass@5* – 77,7% против 67,3% для русского набора, то есть разница составляет около 10 п.п. по обеим метрикам. Даже при наилучших найденных параметрах для каждого языка преимущество англоязычной постановки сохраняется – 77,7% против 67,6% для русского по *pass@5*, что соответствует разнице в 10,1 п.п. Для английского набора наилучшей температурой оказалась 0,6, при которой были получены значения *pass@1* – 82,3% и pass@5 – 77,7%. Для русского набора лучшей температурой оказалась 0,8, при которой достигнуты *pass@1* – 72,6 % и pass@5 – 67,6%. При этом повышение температуры до 1,0 приводит к резкому ухудшению качества. Для английского набора *pass@1* снижается с 82,3% до 60,4%, *pass@5* – с 77,7% до 63,3%, а число ошибок возрастает с 212 до 366; для русского набора *pass@1* падает с 72,6% до 41,5%, *pass@5* – с 67,6% до 47,4%, а число ошибок увеличивается с 311 до 527. Это показывает, что качество генерации чувствительно к выбору температуры и параметров декодирования, и при неоптимальных настройках очень быстро деградирует. Анализ ошибочных генераций показывает, что преобладают две группы ошибок. Синтаксические ошибки в основном связаны с нарушением структуры программы, незавершёнными строковыми или скобочными конструкциями, ошибками отступов и некорректными границами блоков. Логические ошибки, напротив, чаще соответствуют синтаксически корректным, но неверным реализациям, не проходящим тесты на функциональную корректность; в отдельных случаях они сопровождаются ошибками типов, имён или индексации. При росте температуры число ошибок обеих групп увеличивается, причём для русскоязычных постановок синтаксические сбои выражены сильнее. Анализ алгоритма *top-k* подтвердил, что улучшение возможно при совместной настройке нескольких параметров декодирования. Наилучшие результаты в обоих случаях были получены при $k = 10$. Для английского набора *pass@1* – 83,5%, для русского – 73,3%, что соответствует улучшению на 1,2 п.п. и 0,7 п.п. по сравнению с подбором лишь лучшей температуры или 10,4 п.п. и 14,4 п.п. при сравнении с параметром $k = 1$. Следовательно, ограничение выборки позволяет повысить качество генерации при более тонкой настройке параметра *k* в алгоритме *top-k*.

Заключение

Проведённые эксперименты показали, что качество генерации программного кода моделью Qwen3.5-4B-Base существенно зависит как от языка постановки задачи, так и от параметров декодирования. Для англоязычного набора HumanEval наилучший результат был получен при температуре 0,6, для русскоязычного варианта mHumanEval – при температуре 0,8; при повышении температуры до 1,0 качество на обоих наборах заметно снижается, а

число ошибок возрастает. Дополнительная настройка алгоритма *top-k* позволила получить лучший результат при $k = 10$ и дала прирост качества для обоих языков. Англоязычные постановки во всех сериях эксперимента показали более высокие результаты, чем русскоязычные. Это подтверждает, что в задачах мультиязычной генерации программного кода необходимо учитывать не только выбор модели, но и язык постановки задачи вместе с режимом декодирования. Несмотря на достигнутые улучшения, разница между английскими и русскими постановками остаётся значительной и влияет на вероятность получения корректно исполняемого кода. Полученные результаты могут быть использованы при подборе параметров генерации для практических сценариев генерации программного кода.

Список использованных источников

1. A Survey on Large Language Models for Code Generation / J. Jiang [et al.] // arXiv preprint arXiv:2406.00515v2 – 2024. – URL: https://arxiv.org/pdf/2406.00515.

2. Evaluating large language models trained on code / M. Chen [et al.] // arXiv preprint arXiv:2107.03374. – 2021. – URL: https://arxiv.org/abs/2107.03374v2.

3. Raihan, N. mHumanEval: A Multilingual Benchmark to Evaluate Large Language Models for Code Generation / N. Raihan, A. Anastasopoulos, M. Zampieri // Proc. of the 2025 NAACL: Human Language Technologies. – 2025. – С. 11432–11461. – URL: https://aclanthology.org/2025.naacl-long.570/.

4. Hot or Cold? Adaptive Temperature Sampling for Code Generation with Large Language Models / Y. Zhu [et al.] // arXiv preprint arXiv:2309.02772v3. – 2023. – URL: https://arxiv.org/abs/2309.02772v3.

5. Qwen3.5: Towards Native Multimodal Agents [Электронный ресурс]. – 15.02.2026. – URL: https://qwen.ai/blog?id=qwen3.5 (дата обращения: 30.03.2026).

STUDY OF THE EFFECT OF SAMPLING TEMPERATURE ON THE QUALITY OF CODE GENERATION BY A LARGE LANGUAGE MODEL USING MULTILINGUAL DATA

S.V. Gladkov[1], I.A. Kilbas [1], R.A. Paringer[1]

[1]Samara University, Samara
e-mail: gerrhellpony@gmail.com

Abstract

This paper examines the impact of sampling temperature and the top-k algorithm on the quality of code generation by the Qwen3.5-4B-Base model on the HumanEval dataset and its Russian variant from mHumanEval. Quality is assessed using pass@k metrics for Python tasks. Logical and syntactic errors are also analyzed. It is shown that English tasks consistently yield better results, quality depends significantly on temperature, and additional tuning with the top-k algorithm yields further improvements in the metric.

Keywords: large language models, sampling, temperature, code generation, HumanEval, mHumanEval, Qwen 3.5.