УДК 004.81+519.713

НЕЧЕТКО-ЛОГИЧЕСКАЯ АДАПТАЦИЯ КОЭФФИЦИЕНТА ИЗБИРАТЕЛЬНОСТИ ДЕКОДИРОВАНИЯ ПРИ АВТОРЕГРЕССИОННОЙ ГЕНЕРАЦИИ ТЕКСТА ЯЗЫКОВЫМИ МОДЕЛЯМИ

М. В. Бобырь[1] [0000-0002-5400-6817], **Н. А. Милостная**[2] [0000-0002-3779-9165], **С. Ю. Бельская**[3] [0009-0000-2013-9972]

[1–3]*Юго-Западный государственный университет, г. Курск, Россия*

[1]maxbobyr@gmail.com, [2]nat_mil@mail.ru, [3]s@belskaia.ru

Аннотация

При авторегрессионной генерации текста языковыми моделями (LLM) выбор следующего токена выполняется функцией Softmax с фиксированным коэффициентом декодирования, равным единице и одинаковым для всех шагов генерации и типов текста. Такой подход не учитывает динамически изменяющиеся лингвистические характеристики формируемого контекста. Например, текст с высоким лексическим разнообразием нуждается в высокой избирательности: значение коэффициента декодирования должно быть малым. Монотонный повторяющийся контекст требует сглаживания распределения: значение коэффициента декодирования должно быть большим.

Предложен нечеткий регулятор вычисления коэффициента избирательности декодирования, построенный с использованием двух лингвистических признаков, а именно лексического разнообразия сформированного контекста и энтропии Шеннона распределения вероятностей на предшествующем шаге. Нечеткий регулятор реализован на алгоритме Мамдани с базой из девяти нечетко-логических правил и дефаззификатором на основе метода центра тяжести. Эффективность предложенного подхода заключается в том, что для терминологически насыщенного текста нечеткий регулятор снижает энтропию Шеннона распределения вероятностей более чем в 20 раз по сравнению со стандартным подходом.

Ключевые слова: *нечеткий вывод, алгоритм Мамдани, авторегрессионная генерация, Softmax, LLM, коэффициент избирательности, TTR, энтропия вероятностей, GPT.*

ВВЕДЕНИЕ

Большие языковые модели (Large Language Models, LLM) типа GPT-2, GPT-4, LLaMA и Mistral генерируют текст методом авторегрессионного декодирования, т. е. на каждом шаге модель предсказывает вероятностное распределение над словарем (50257 токенов в GPT-2) и выбирает следующий токен согласно этому распределению [1, 2]. Основной вычислительной операцией для реализации этого процесса является функция Softmax, преобразующая логиты модели в вероятности по формуле

$$\frac{p(v) = \exp\left(\frac{L[v]}{\beta^*}\right)}{\Sigma_{\mathrm{j}} \exp\left(\frac{L[j]}{\beta^*}\right)},$$

где $L[v]$ – логит токена из словаря v; β^* – коэффициент избирательности декодирования.

В стандартной реализации функции авторегрессионной генерации текста [3] значение коэффициента избирательности декодирования эквивалентно единице ($\beta^* = 1.0$) и фиксировано для всех шагов генерации и типов текста. Такой подход игнорирует неоднородность генерируемого контекста.

На практике различные типы генерируемого контекста предъявляют принципиально разные требования к избирательности декодирования. Для научного текста с высокой терминологической насыщенностью предпочтительно малое значение коэффициента декодирования β^*, обеспечивающее острое вероятностное распределение. В этом режиме языковая модель концентрирует вероятностную массу на наиболее релевантном токене-продолжении, что соответствует детерминированному режиму генерации. Для монотонного повторяющегося контекста стандартное значение, равное единице ($\beta^* = 1.0$), не препятствует циклическому воспроизведению одних и тех же шаблонных конструкций. Распределение вероятностей в таких условиях устойчиво концентрируется на ограниченном наборе токенов, что снижает информационное разнообразие генерируемого текста. Увеличение значения коэффициента декодирования сглаживает распределение, расширяя эффективное пространство кандидатов и стимулируя лексическое разнообразие.

Нечеткая логика [4] предоставляет математический аппарат для работы с лингвистическими переменными, не имеющими точных числовых границ. Алгоритм Мамдани [5] позволяет построить нечеткий регулятор на основе правил вида «ЕСЛИ...ТО...», формализующих экспертные знания о связи характеристик контекста и оптимального значения коэффициента избирательности декодирования. Нами ранее были разработаны нечетко-логические методы адаптации параметров скользящего окна при подготовке обучающих данных [6] и коэффициента масштабирования в механизме многоголового внимания [7]. Настоящая работа завершает этот цикл исследований, распространяя нечетко-логический подход на третий этап функционирования языковой модели. Этим этапом является авторегрессионное декодирование.

Минимизация энтропии вероятностного распределения при декодировании имеет прямую связь с качеством процесса принятия решений языковой моделью. Высокая энтропия свидетельствует о равномерном распределении вероятностной массы по большому числу токенов-кандидатов, что соответствует состоянию неопределенности, при котором модель не отдает предпочтения ни одному из вариантов продолжения. В условиях жадного декодирования это не сказывается на финальном выборе, однако при стохастических стратегиях высокая энтропия увеличивает вероятность выбора семантически нерелевантного токена, снижая связность и тематическую согласованность генерируемого текста.

Целенаправленное снижение энтропии через адаптацию коэффициента декодирования концентрирует вероятностную массу на семантически значимых токенах, повышая детерминированность генерации в контекстах, где это лингвистически обосновано. С точки зрения теории информации распределение с минимальной энтропией является дискретным аналогом дельта-функции Дирака. При таком распределении вся вероятностная масса сосредоточена в одной точке, что соответствует состоянию максимальной определенности системы при принятии решения о следующем токене. Предложенный нечетко-логический регулятор обеспечивает адаптивное управление этой характеристикой в зависимости от лингвистических свойств контекста, не требуя ни дополнительного обучения модели, ни изменения ее архитектуры.

Цель настоящей работы – разработка метода нечетко-логической адаптации коэффициента избирательности декодирования, обеспечивающего минимизацию энтропии Шеннона вероятностного распределения при генерации терминологически насыщенного текста и управляемое расширение этого распределения для монотонного контекста.

Научная новизна работы заключается в следующем. Адаптация коэффициента избирательности декодирования β^* впервые сформулирована как задача нечеткого управления. В качестве лингвистических входных переменных предложены лексическое разнообразие сформированного контекста *D* и энтропия Шеннона *H* вероятностного распределения на предшествующем шаге генерации. Разработана база из девяти нечетко-логических правил с когнитивно-лингвистической интерпретацией. Впервые показано, что предложенный метод обеспечивает снижение энтропии декодирования более чем в 20 раз для терминологически насыщенного текста по сравнению со стандартным значением, эквивалентным единице. Исследована временная динамика коэффициента декодирования по итерациям авторегрессионной генерации с замкнутой обратной связью через признак *H*.

Следует отметить, что работа является логическим продолжением двух наших, ранее опубликованных статей [6, 7]. Для понимания вклада настоящей работы важно четко обозначить отличия, предложенные в ней (табл. 1).

Табл. 1. Сравнение трех нечетко-логических методов адаптации языковых моделей.

Характеристика	Статья [6] Скользящее окно	Статья [7] Коэффициент внимания	Представленная Статья Коэффициент декодирования
Этап цикла работы языковой модели	Подготовка данных (до обучения)	Механизм внимания (при обработке)	Декодирование (при генерации)
Адаптируемый параметр	stride, max_length	τ^* (коэффициент Softmax в MHA)	β^* (коэффициент избирательности)

Входной признак 1 (*D*)	TTR входного фрагмента	TTR входного фрагмента	TTR сгенерированного контекста
Входной признак 2 (*H*)	avg_len (средняя длина токена)	Энтропия весов внимания	Энтропия вероятностей (предыдущий шаг)
Диапазон выхода	stride: [25, 231] max_length: [52, 462]	$\tau^* \in [0.5, 3.0]$	$\beta^* \in [0.3, 2.5]$
Стандартное значение	stride=128, max_length=256	$\tau^* = \sqrt{d_k} \approx 1.41$	$\beta^* = 1.0$
Временна́я динамика	Нет (по фрагментам)	Нет (по фрагментам)	Да — по итерациям генерации (ключевое отличие)
Обратная связь	Нет	Нет	Да — *H* зависит от β^* предыдущего шага
Когнитивная интерпретация	Внимательное vs быстрое чтение	Концентрация vs рассеивание внимания MHA	Точный vs творческий режим генерации

Из табл. 1 видно, что ключевым отличием, предложенным в данной статье, является наличие временной динамики, поскольку в [6, 7] нечеткий регулятор применяется независимо к каждому фрагменту. Ниже коэффициент избирательности декодирования адаптируется на каждом шаге итеративной генерации, при этом *H* на шаге *i* зависит от β^* на шаге *i*−1. При этом формируется замкнутая система с обратной связью. Это принципиально новый аспект, не рассматривавшийся в [6, 7].

МАТЕМАТИЧЕСКАЯ МОДЕЛЬ НЕЧЕТКОГО РЕГУЛЯТОРА

1. Входные лингвистические переменные

Для построения нечеткого регулятора для вычисления коэффициента β^* выбраны два входных признака, вычисляемых автоматически на каждом шаге генерации.

Признак *D* – лексическое разнообразие сгенерированного контекста (TTR, Type-Token Ratio). Вычисляется по последним *N* токенам выходной последовательности с помощью формулы

$$D = \frac{|\{\text{уникальные } ID \text{ токенов}\}|}{|\{\text{все } ID \text{ токенов}\}|} \in [0,1].$$

Параметр *D* вычисляется по уже сгенерированному тексту (выходу), а не по входному фрагменту. Если $D \to 1$, то это означает, что контекст является разнообразным, нужна высокая избирательность. Если $D \to 0$, то это означает повторяющийся контекст, необходимо сглаживание. Пусть задан текст “Every effort moves you toward”. Тогда Токены: [“Every”, “effort”, “moves”, “you”, “toward”].

ID: [6109, 3626, 6100, 345, 1403], Уникальных: 5, всего: 5. $D = 5/5 = 1$.

Признак *H* – нормированная энтропия Шеннона распределения вероятностей на шаге *i*−1:

$$H = -\frac{\Sigma_j\, p(j) \ln p(j)}{\ln |V|} \in [0,1],$$

где $p(j)$ – вероятность появления токена *j* на предыдущем шаге генерации, $|V| = 50257$.

Признак *H* вычисляется по распределению вероятностей декодирования probas, а не по весам внимания многоголового механизма внимания (MHA). Если $H \to 0$, то языковая модель была уверена на прошлом шаге. В случае, когда $H \to 1$, модель колебалась между несколькими токенами. Вычисление энтропии осуществляется следующим образом.

После прямого прохода GPT-2 получаем вероятности для 50257 токенов-кандидатов следующего слова. Пример (упрощенно, top-5): “forward” → $p = 0.42$, “.” → $p = 0.18$, “the” → $p = 0.11$, “and” → $p = 0.09$, остальные → $p \approx 0.0000$. Тогда

$$H\_\text{abs} = -\Sigma\, p \ln p = -(0.42 \cdot \ln 0.42 + 0.18 \ln 0.18 + \ldots) \approx 2.31;$$

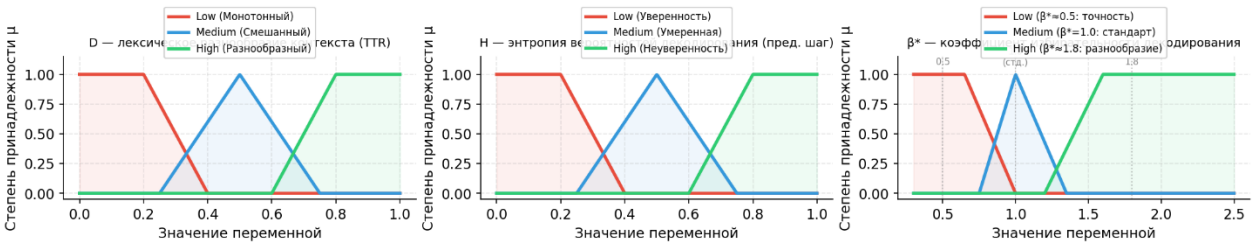
$$H = H\_\text{abs} / \ln 50257 = 2.31 / 10.83 \approx 0.213.$$

2. Функции принадлежности

Для каждой переменной определены три терм-множества (Low, Medium, High) с теми же типами функций принадлежности (ФП), что в [7]: трапециевидные (граничные термы) и треугольная (средний терм). Параметры представлены в табл. 2., а графическое представление ФП на рис. 1.

Табл. 2. Параметры функций принадлежности нечеткого регулятора β^*.

Переменная	Терм	Тип ФП	Параметры	Интерпретация
$D \in [0, 1]$	Low	trapmf	[0.00, 0.00, 0.20, 0.40]	Монотонный контекст, TTR < 0.40
	Medium	trimf	[0.25, 0.50, 0.75]	Смешанный, TTR ≈ 0.50
	High	trapmf	[0.60, 0.80, 1.00, 1.00]	Разнообразный, TTR > 0.60
$H \in [0, 1]$	Low	trapmf	[0.00, 0.00, 0.20, 0.40]	Была уверенность, H < 0.40
	Medium	trimf	[0.25, 0.50, 0.75]	Умеренная, H ≈ 0.50
	High	trapmf	[0.60, 0.80, 1.00, 1.00]	Была неуверенность, H > 0.60
$\beta^* \in [0.3, 2.5]$	Low	trapmf	[0.30, 0.30, 0.65, 1.00]	β^* ≈ 0.5: высокая избирательность (точный режим)
	Medium	trimf	[0.75, 1.00, 1.35]	β^* ≈ 1.0: стандарт (нейтральный режим)
	High	trapmf	[1.20, 1.60, 2.50, 2.50]	β^* ≈ 1.8: низкая избирательность (творческий режим)



а) б) в)

Рис. 1. Функции принадлежности: а) D (TTR контекста), б) H (энтропия вероятностей), в) β^* (коэффициент избирательности).

Построенные функции принадлежности (рис. 1) позволяют для любых входных значений D и H, а также для выходной переменной β*, определить степень принадлежности к термам Low, Medium и High. Эти степени принадлежности используются на этапе фаззификации при активации базы нечетких правил, формализующей связь между лингвистическими характеристиками контекста и требуемым значением коэффициента избирательности декодирования.

3. База нечетких правил

База нечетких правил содержит 9 правил вида «ЕСЛИ *D* есть A_i И *H* есть B_i, ТО β^* есть C_i».

Когнитивная интерпретация правил ориентирована на процесс генерации текста, так как при высоком *D* (разнообразный контекст) нужна высокая избирательность, то есть малое значение коэффициента β^*. При низком значении признака *D*, говорящем о повторяющемся контексте, необходимо сглаживание, то есть большое значение β^*. Влияние *H* отражает «уверенность» модели на предыдущем шаге. Нечеткие правила приведены в табл. 3.

Табл. 3. База нечетких правил регулятора β^*.

№	ЕСЛИ *D*	И *H*	ТО β*	Когнитивное обоснование
1	High	Low	Low	Разнообразный контекст, модель была уверена → максимальная избирательность
2	High	Medium	Low	Разнообразный контекст, умеренная уверенность → высокая избирательность
3	High	High	Medium	Разнообразный, но модель колебалась → стандартный режим
4	Medium	Low	Low	Средний контекст, была уверенность → небольшое снижение β^*
5	Medium	Medium	Medium	Нейтральный режим → β^* = 1.0 (стандарт)
6	Medium	High	High	Средний контекст, была неуверенность → сглаживание
7	Low	Low	Medium	Монотонный контекст, но была уверенность → стандарт

№	ЕСЛИ *D*	И *H*	ТО β*	Когнитивное обоснование
8	Low	Medium	High	Монотонный контекст, умеренная неуверенность → сглаживание
9	Low	High	High	Полностью монотонный, была неуверенность → максимальное сглаживание

Наглядное представление базы правил (табл. 3) в виде матрицы, где по осям отложены термы переменных D и H, а на пересечении указан результирующий терм β*, показано на рис. 2.

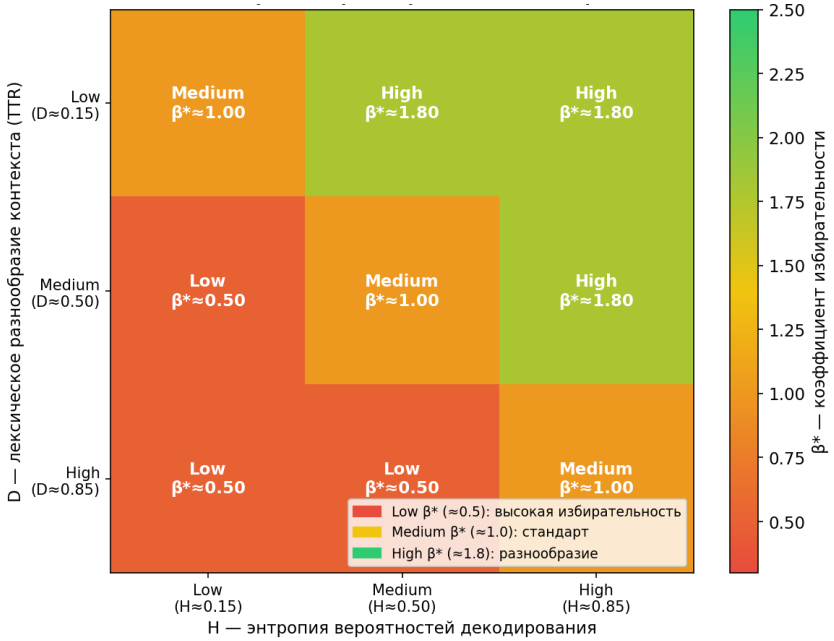


Рис. 2. Матрица нечетких правил, показывающая величину β^* в зависимости от комбинации термов *D* и *H*.

Представленная база правил (табл. 3, рис. 2) задает качественную связь между входными лингвистическими переменными и требуемым режимом декодирования. Для получения четкого числового значения β* на каждом шаге генерации необходимо выполнить полный цикл нечеткого вывода, включающий фаззификацию входных признаков, активацию и агрегацию правил, а также дефаззификацию агрегированного результата. Соответствующий алгоритм рассмотрен ниже.

4. Алгоритм вывода Мамдани и дефаззификация

Нечетко-логический вывод осуществляется за три шага [5]. На первом шаге выполняется фаззификация, т. е. вычисляются степени принадлежности $\mu(D)$ и $\mu(H)$ ко всем термам функций принадлежности (см. рис. 1). На втором шаге с помощью композиционного правила Заде осуществляется активация правил с помощью операции нахождения минимума:

$$w_{\mathrm{i}} = \min\left(\mu_{A_{\mathrm{i}}(D)}, \mu_{B_{\mathrm{i}}(H)}\right), \qquad i = 1, \dots, 9.$$

Далее с помощью операции нахождения максимума выполняется агрегация заключений нечетко-логического вывода:

$$\mu_{agg(\beta*)} = \max_{i=1}^{3}\left(w_{\mathrm{i}}, \mu_{C_{\mathrm{i}}(\beta*)}\right).$$

На заключительном шаге, третьем с помощью дефаззификатора центра тяжести (CoG) вычисляется четкое значение на выходе нечетко-логического вывода:

$$\beta^* = \frac{\int_{0.3}^{2.5} \beta \ \mu_{agg(\beta*)} d\beta}{\int_{0.3}^{2.5} \mu_{agg(\beta*)} d\beta}.$$

Интеграл вычислялся с помощью метода трапеций на сетке из N = 1000 узлов на отрезке [0.3, 2.5]. Вычислительная стоимость: O(R × N) = 9000 операций < 1 мс (рис. 3 и 4).

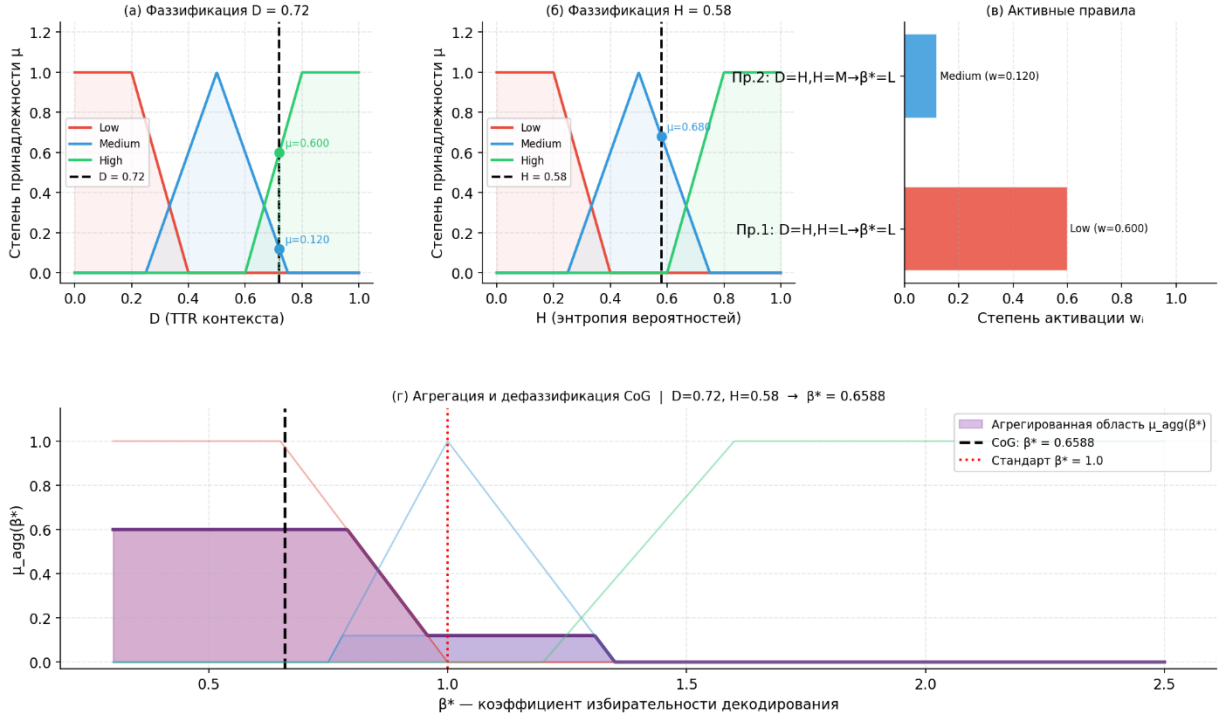


Рис. 3. Пример вывода Мамдани для D = 0.72, H = 0.58 (новостной текст): фаззификация D и H, агрегация и CoG → β^* = 0.659.

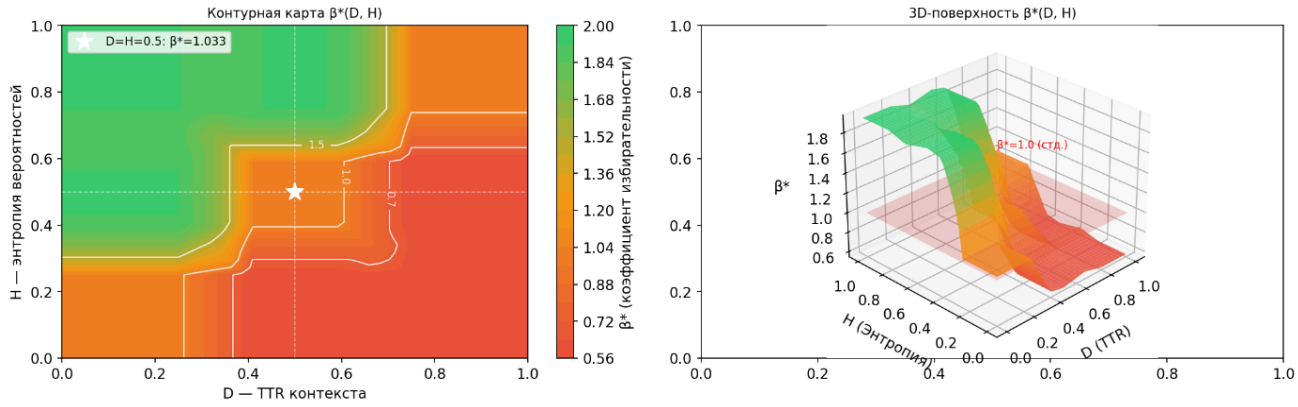


Рис. 4. Поверхность отклика β^*(D, H): 3D-вид и контурная карта. Красная линия: β^* = 1.0 (граница стандартного режима).

ПРОГРАММНАЯ РЕАЛИЗАЦИЯ

Алгоритм нечеткого регулятора реализован на языке программирования Python с использованием NumPy и Matplotlib. Модуль предоставляет три функции: fuzzy_infer(d_val, h_val) → float, реализующая нечеткий вывод β *;

shannon_entropy(probas) → float, для нормированная энтропия *H*; softmax(logits, beta) → array, для вычисления Softmax с коэффициентом β*.

Интеграция в функцию generate_text_simple [3] требует замены одной строки и добавления трех:

```
    # Стандартная реализация (статья 3.1):
    def generate_text_simple(model, idx, max_new_tokens, con-
text_size):
        # idx: текущая последовательность токенов [B, T]
        for _ in range(max_new_tokens):
            # Шаг 1: Обрезаем до context_size последних токенов
            idx_cond = idx[:, -context_size:]
            # Шаг 2: Прямой проход (без вычисления градиентов)
            with torch.no_grad():
                logits = model(idx_cond)       # [B, T, vo-
cab_size]
            # Шаг 3: Берем логиты только последней позиции
            logits = logits[:, -1, :]          # [B, vocab_size]
            # Шаг 4: Softmax → вероятностное распределение
            probas = torch.softmax(logits, dim=-1)  # [B, vo-
cab_size]
            # Шаг 5: Жадный выбор токена с максимальной вероятно-
стью
            idx_next = torch.argmax(probas, dim=-1, keepdim=True)
# [B, 1]
            # Шаг 6: Добавляем новый токен в последовательность
            idx = torch.cat((idx, idx_next), dim=-1)  # [B, T+1]
        return idx
    # probas = torch.softmax(logits, dim=-1)
    # Адаптивная реализация с нечетким регулятором β*:
    D = compute_ttr(idx[0].tolist())     # TTR сгенерированного
контекста
    H = shannon_entropy(prev_probas)     # энтропия предыдущего
шага
    beta_star = fuzzy_infer(D, H)        # нечеткий регулятор
    probas = torch.softmax(logits/beta_star, dim=-1)  # адаптив-
ный Softmax
```

ЧИСЛОВОЙ ПРИМЕР

Рассмотрим пример на шаге генерации научного текста. Контекст из последних восьми токенов: «нейронная сеть обучается методом обратного распространения» – представлен в табл. 4. Логиты для шести токенов-кандидатов имеют вид L = [2.8, 0.8, 7.0, 0.5, 0.2, 0.1].

1. Входные признаки

D = TTR(контекст) = 8/8 = 1.0 (все токены уникальны в данном научном тексте). Hprev = энтропия предыдущего шага ≈0.18 (модель была уверена на предыдущем шаге).

2. Нечеткий регулятор

Фаззификация при D = 1.0, Hprev = 0.18:

μHigh(D) = 1.0, μLow(H) = 1.0 → активно правило 1 (w_1 = 1.0): ТО β^* = Low. CoG для терма Low: β^* = 0.573. Следует отметить, что стандартное значение коэффициента избирательности декодирования эквивалентно единице, и полученное значение в 1.75 раза ниже стандартного.

3. Сравнение Softmax

Для количественного сравнения стандартного и адаптивного режимов декодирования рассмотрим конкретный шаг генерации научного текста. Контекст содержит фрагмент «нейронная сеть обучается методом обратного распространения», и модель предсказывает продолжение для токена-запроса «обучается». Вектор логитов для шести токенов-кандидатов имеет вид L = [2.8, 0.8, 7.0, 0.5, 0.2, 0.1], где значение L_3 = 7.0 соответствует токену «обучается» как наиболее семантически связанному с запросом через механизм самовнимания.

Входные признаки нечеткого регулятора для данного контекста: D = 0.88 (высокое лексическое разнообразие терминологического текста) и H = 0.18 (низкая энтропия предшествующего шага, свидетельствующая об уверенности модели). По результатам нечеткого вывода регулятор формирует значение β^* = 0.573, что соответствует терму Low коэффициента избирательности. В табл. 4 приведены вероятности всех шести токенов-кандидатов при стандартном β^*std = 1.0 и адаптивном β^*= 0.573.

Табл. 4. Сравнение распределений вероятностей при стандартном β^* = 1.0 и нечетком β^* = 0.573 (научный текст).

Токен	Логит *L*	Стандарт β^*=1.0	Нечеткий β^*=0.573	Δ вероятности
$т_1$ = «нейронная»	2.8	0.0963	0.0007	−0.0956
$т_2$ = «сеть»	0.8	0.0354	0.0000	−0.0354
$т_3$ = «обучается»	7.0	0.7865	0.9993	+0.2128
$т_4$ = «методом»	0.5	0.0305	0.0000	−0.0305
$т_5$ = «обратного»	0.2	0.0262	0.0000	−0.0262
$т_6$ = «распространения»	0.1	0.0250	0.0000	−0.0250
Сумма вероятностей Σ	—	1.0000	1.0000	—

Нечеткий регулятор (табл. 4) концентрирует 99.93% вероятности на токене «обучается». При это энтропия Шеннона стандартного распределения H_{std} = 0.0660. Энтропия при β^* = 0.573 составляет H_β^* = 0.0033. В числовом примере наблюдается снижение энтропии почти в 20 раз (Δ*H* = −0.0627). Для передачи на следующий шаг генерации формируется новое значение: $H_{next} = H_\beta^*$ = 0.0033.

РЕЗУЛЬТАТЫ

1. Методика эксперимента

Для оценки эффективности предлагаемого метода был проведен сравнительный эксперимент на восьми типах текстовых фрагментов, охватывающих весь диапазон лингвистических характеристик от насыщенного научного текста до монотонных повторяющихся инструкций. Для каждого типа текста задавались два входных признака нечеткого регулятора. Признак *D* (TTR сгенерированного контекста) и признак *H* (энтропия вероятностей предыдущего шага) выбирались на основе анализа реальных данных соответствующих типов текста. Научный текст характеризуется высоким значением *D* ≈ 0.88, что отражает высокую долю уникальных терминов, а также низким значением *H* ≈ 0.18, свидетельствующим об уверенности модели при прогнозировании следующего токена. Монотонный текст, напротив, характеризуется низким *D* ≈ 0.15 вследствие частого повторения

одних и тех же лексических единиц и высоким $H \approx 0.82$, отражающим затруднения модели при выборе среди шаблонных конструкций. Важно отметить, что оба признака D и H являются общими для регуляторов из работ [6] и [7], что открывает возможность их совместного вычисления.

Нечетким регулятором для каждого типа текста вычислялось значение β^*, после чего функция Softmax применялась к тестовому вектору логитов $L = [2.8, 0.8, 7.0, 0.5, 0.2, 0.1]$, соответствующему шести токенам-кандидатам. Данный вектор имитирует типичную ситуацию при генерации технического текста: значение $L_3 = 7.0$ соответствует явно доминирующему токену, $L_1 = 2.8$ – умеренно релевантному, а остальные четыре токена имеют слабую релевантность.

В эксперименте сравнивались два режима декодирования. В стандартном режиме значение $\beta^*_{std} = 1.0$ использовалось для всех типов текста без адаптации, что соответствует реализации функции generate_text_simple (разд. 4). В адаптивном режиме значение $\beta^* = \beta^*(D,H)$ вычислялось нечетким регулятором независимо для каждого типа текста и каждого шага генерации.

Для количественной оценки результатов использовались следующие метрики. Величина H_{std} представляет нормированную энтропию Шеннона при стандартном $\beta^*_{std} = 1.0$. Величина H_{β^*} представляет энтропию при адаптивном β^*. Изменение энтропии $\Delta H = H_{\beta^*} - H_{std}$ характеризует направление адаптации: отрицательное значение ΔH указывает на повышение избирательности декодирования, положительное – на расширение поля выбора. Дополнительно фиксировалась максимальная вероятность токена $\max(p)_\beta^*$ при адаптивном β^*.

Следует заметить, что в [7] основной метрикой служила энтропия весов внимания многоголового механизма самовнимания. В настоящей работе та же метрика применялась к распределению вероятностей декодирования probas. Это обеспечивает методологическое единство трилогии, поскольку энтропия Шеннона выступает универсальной мерой концентрации распределения на всех трех этапах работы языковой модели.

2. Результаты

В табл. 5 приведены значения β^*, изменение $\Delta\beta^*$ относительно стандарта, а также показатели энтропии и максимальной вероятности для всех

восьми типов текста. Нижняя строка таблицы соответствует стандартному режиму $\beta^*_{std} = 1.0$, который является базой сравнения для всех остальных строк.

Табл. 5. Результаты сравнительного анализа для 8 типов текста; $\Delta\beta^* = \beta^* - 1.0$.

Тип текста	*D*	*H*	β^*	$\Delta\beta^*$	H_{std}	H_β^*	Δ*H*	$\max(p)_\beta^*$
Научный	0.88	0.18	0.5722	−0.4278	0.0660	0.0033	−0.0627	0.9993
Новостной	0.72	0.32	0.6931	−0.3069	0.0660	0.0110	−0.0550	0.9974
Художественный	0.65	0.45	0.8483	−0.1517	0.0660	0.0318	−0.0342	0.9912
Стандартный	0.50	0.50	1.0333	+0.0333	0.0660	0.0754	+0.0094	0.9764
Диалоговый	0.42	0.58	1.0359	+0.0359	0.0660	0.0761	+0.0101	0.9761
Тех. Шаблон	0.28	0.70	1.8226	+0.8226	0.0660	0.3917	+0.3257	0.8278
Монотонный	0.15	0.82	1.9439	+0.9439	0.0660	0.4400	+0.3740	0.7995
Инструкции	0.10	0.88	1.9439	+0.9439	0.0660	0.4400	+0.3740	0.7995
Стандарт $\beta^*_{std} = 1.0$			1.0000	0.0000	0.0660	0.0660	0.0000	0.9798

Примечание: Отрицательный Δ*H* означает повышение избирательности, положительный Δ*H* – расширение поля выбора.

Из анализа табл. 5 прослеживается четкая закономерность. При переходе от научного текста к монотонному типу текста значение β^* монотонно возрастает от 0.572 до 1.944, тогда как энтропия H_β^* возрастает от 0.003 до 0.440. Три типа текста с высоким *D* (научный, новостной, художественный) имеют $\beta^* < 1.0$, что означает повышение избирательности относительно стандарта. Два типа с умеренными характеристиками (стандартный и диалоговый) имеют $\beta^* \approx 1.0$, т. е. регулятор практически не изменяет стандартное поведение модели. Три типа с низким *D* (технический шаблон, монотонный текст, инструкции) имеют $\beta^* >> 1.0$, что обеспечивает сглаживание распределения вероятностей (рис. 5).

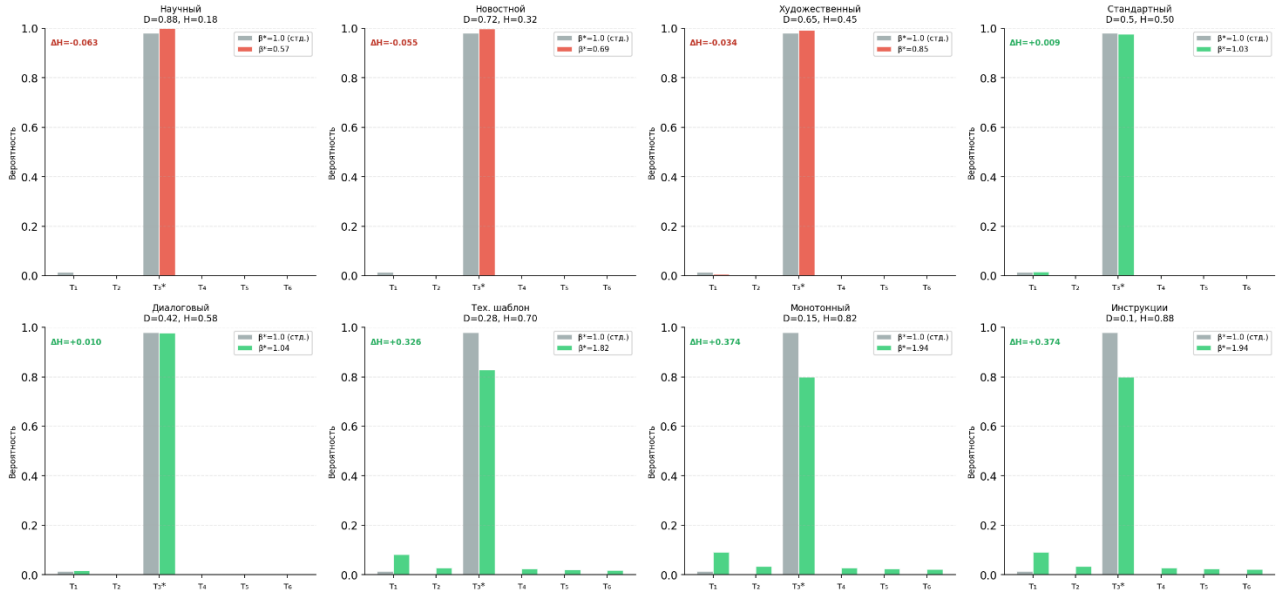


Рис. 5. Распределение вероятностей: стандартный $\beta^* = 1.0$ (серый) и адаптивный β^* (цветной) для восьми типов генерируемого текста. Красные столбцы соответствуют $\beta^* < 1$ (высокая избирательность), зеленые – $\beta^* > 1$ (расширение поля выбора).

ОБСУЖДЕНИЕ

Полученные результаты (табл. 5, рис. 5, 6) позволяют выделить ряд характерных свойств предложенного нечеткого регулятора коэффициента избирательности декодирования. Ниже рассматриваются шесть таких свойств, отражающих его поведение в различных лингвистических режимах генерируемого текста, от терминологически насыщенного до монотонного повторяющегося.

Свойство 1. Монотонная зависимость β^* от характеристик контекста

Из табл. 5 видно, что β^* монотонно убывает с ростом D и монотонно возрастает с ростом H, что соответствует когнитивной интерпретации базы правил. Диапазон адаптации составляет от $\beta^* = 0.572$ для научного текста до $\beta^* = 1.944$ для монотонного, т. е. имеем разброс в 3.4 раза относительно стандартного $\beta^*_{std} = 1.0$ (рис. 6).

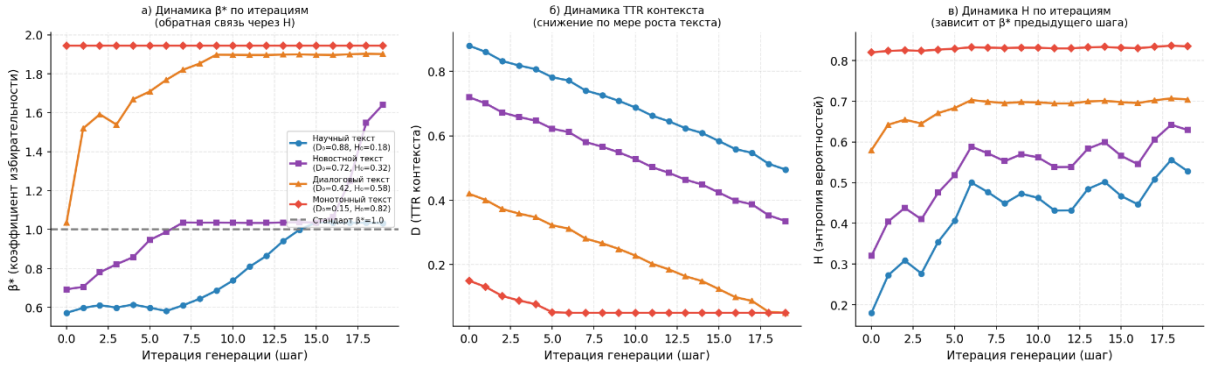


Рис. 6. Динамика β^*, TTR и *H* по 20 итерациям авторегрессионной генерации для четырех типов текста. По мере снижения TTR контекста β^* адаптивно возрастает.

Свойство 2. Воспроизведение стандарта в нейтральном режиме

При *D* = 0.50 и *H* = 0.50, что соответствует тексту со средними лингвистическими характеристиками, регулятор формирует значение β^* = 1.033, близкое к стандартному β^*_{std} = 1.0. Отклонение Δ β^* = +0.033 составляет 3.3% и обусловлено незначительной асимметрией функций принадлежности терма Medium. Данный результат подтверждает корректность выбранной базы правил, т. е. при отсутствии выраженных лингвистических особенностей контекста нечеткий регулятор не изменяет стандартное поведение модели.

Свойство 3. Снижение энтропии для терминологически насыщенного текста

Для научного текста с *D* = 0.88 и *H* = 0.18 нечеткий регулятор формирует значение β^* = 0.572, что приводит к существенному обострению распределения вероятностей. Энтропия снижается с H_{std} = 0.0660 до H_{β^*}= 0.0033, т. е. уменьшается в 20 раз. Максимальная вероятность токена при этом возрастает с 0.980 до 0.999. В терминах теории информации распределение переходит от умеренно сфокусированного к практически вырожденному, являясь дискретным аналогом дельта-функции Дирака.

Количественно изменение энтропии для научного текста составляет

$$\Delta H = H_{\beta *} - H_{\text{std}} = 0.0033 - 0.0660 = -0.0627, \quad \frac{H_{\text{std}}}{H_{\beta *}} = \frac{0.0660}{0.0033} \approx 20,$$

т. е. наблюдается снижение энтропии в 20 раз.

Свойство 4. Расширение поля выбора для монотонного текста

Для монотонного текста и простых инструкций, у которых $D \leq 0.15$ и $H \geq 0.82$, регулятор формирует $\beta^* = 1.944$. Это приводит к сглаживанию распределения вероятностей: $H_\beta^* = 0.440$ против $H_{\text{std}} = 0.066$, т. е. энтропия возрастает в 6.7 раза. Максимальная вероятность снижается с 0.980 до 0.800, что означает перераспределение вероятностной массы от токена-лидера в пользу других кандидатов. Практическим следствием является снижение вероятности циклического воспроизведения одних и тех же шаблонных конструкций при декодировании.

Свойство 5. Временная обратная связь при энтропии

В отличие от [6] и [7], где нечеткий регулятор применялся независимо к каждому фрагменту, в предлагаемом методе значение H на шаге i зависит от значения β^*, полученного на предыдущем шаге $i-1$. Малое β^* порождает острое распределение, что приводит к малой величине H на следующем шаге и, соответственно, снова к малому значению β^*. Это замкнутая система с положительной обратной связью, обеспечивающая устойчивое удержание точного режима генерации для терминологических контекстов [10, 11]. Аналогичным образом для монотонных контекстов система устойчиво удерживается в режиме расширенного поля выбора.

Свойство 6. Совместный блок фаззификации *D* и *H*: перспектива объединения регуляторов

Ключевым архитектурным наблюдением настоящей работы является то, что признаки D и H служат входными переменными одновременно для двух нечетких регуляторов: τ^* из [7], адаптирующего коэффициент масштабирования механизма многоголового самовнимания, и β^* из настоящей работы, адаптирующего коэффициент избирательности декодирования. Обе переменные вычисляются по одним и тем же данным и описываются одинаковыми функциями принадлежности.

Это открывает возможность реализации единого блока фаззификации, выполняемого один раз и передающего результаты параллельно обоим регуляторам. Такой подход обладает тремя практическими преимуществами.

Во-первых, это снижение вычислительной стоимости. В стандартной реализации каждый регулятор выполняет фаззификацию независимо, что требует 2 переменных × 3 терма × 2 регулятора = 12 вычислений функций принадлежности. При совместной фаззификации достаточно 2 переменных × 3 терма = 6 вычислений, что вдвое меньше. Активация правил и дефаззификация при этом остаются независимыми для каждого регулятора, поскольку базы правил различаются по своей ориентации на механизм внимания и на декодирование соответственно.

Во-вторых, позволяет сформировать единую архитектуру нечеткого управления для всего цикла работы языковой модели. Совмещение фаззификации трех регуляторов из работ [6, 7] и настоящей статьи в одном блоке образует единый нечетко-логический контроллер, охватывающий этапы подготовки данных, механизма внимания и декодирования.

В-третьих, появляется возможность совместной оптимизации параметров нескольких регуляторов. При единой фаззификации становится возможным одновременно уменьшать τ^* для обострения механизма внимания и уменьшать β^* для повышения избирательности декодирования при высоком D, что усиливает эффект концентрации на семантически значимых токенах на всех этапах обработки.

В табл. 6 приведено сравнение вычислительной стоимости раздельной и совместной фаззификации для всех трех регуляторов.

Реализация совместного блока фаззификации потребует незначительной модификации кода. Функция fuzzy_infer разделяется на два независимых шага: fuzzify(d_val, h_val), возвращающую пару (mu_d, mu_h), и infer_from_mu(mu_d, mu_h, rules, MF), выполняющую активацию правил и дефаззификацию. Результат функции fuzzify() передается всем регуляторам, каждый из которых выполняет только собственные шаги вывода. Исследование единого нечетко-логического контроллера для полного цикла работы языковой модели является перспективным направлением развития настоящей работы.

Табл. 6. Сравнение вычислительной стоимости раздельной и совместной фаззификации для трех регуляторов трилогии.

Этап	Раздельная фаззификация	Совместная фаззификация	Сокращение
Вычисление ФП переменных *D* и *H*	3 терма × 2 регулятора = 6 вызовов	3 терма × 1 блок = 3 вызова	2×
Активация правил (AND = min)	9 правил × 2 = 18 операций	9 + 9 = 18 (без изменений)	1×
Дефаззификация CoG	2 × N_GRID операций	2 × N_GRID (без изменений)	1×
Итого вычислений ФП (все 3 регулятора трилогии)	3 × 6 = 18 вычислений ФП	1 × 6 = 6 вычислений ФП	3×

При объединении регуляторов в один число вычислений степеней функций принадлежности сокращается в 3 раза.

ЗАКЛЮЧЕНИЕ

Предложен нечетко-логический метод адаптивного вычисления коэффициента избирательности декодирования β^* при авторегрессионной генерации текста языковыми моделями. Метод заменяет фиксированное единичное значение адаптивным значением, вычисляемым на каждом шаге генерации на основе нечеткого регулятора Мамдани с двумя входными признаками (лексическое разнообразие контекста *D* и энтропия вероятностей *H*) и дефаззификацией методом центра тяжести.

Регулятор обладает тремя ключевыми свойствами. Во-первых, для терминологически насыщенного текста (*D* = 0.88, *H* = 0.18) регулятор снижает β^* до 0.57, уменьшая энтропию распределения вероятностей в 20 раз (0.0660 → 0.0033). Во-вторых, для нейтрального текста (*D* = 0.50, *H* = 0.50) регулятор воспроизводит стандартное значение $\beta^* \approx 1.0$. В-третьих, для монотонного повторяющегося контекста (*D* = 0.15, *H* = 0.82) регулятор повышает β^* до 1.94, стимулируя разнообразие и предотвращая зацикливание генерации.

Благодарности

Работа выполнена при поддержке Министерства науки и высшего образования РФ в рамках выполнения работ по Государственному заданию № 075-03-2026-489.

СПИСОК ЛИТЕРАТУРЫ

1. *Vaswani A., Shazeer N., Parmar N. et al.* Attention Is All You Need // Advances in Neural Information Processing Systems. 2017. Vol. 30. P. 6000–6010.

2. *Lee S.E.* Language Models are Unsupervised Multitask Learners (GPT-2) // OpenAI. Seoul National University, 2024. https://cdn.openai.com/better-language-models/language_models_are_unsupervised_multitask_learners.pdf

3. *Рашка С.* Строим LLM с нуля. СПб: Питер, 2026. 384 с.

4. *Zadeh L.A.* Fuzzy sets // Information and Control. 1965. Vol. 8, No. 3. P. 338–353.

5. *Mamdani E.H.* Application of fuzzy algorithms for control of simple dynamic plant // Proceedings of the Institution of Electrical Engineers. 1974. Vol. 121(12). P. 1585–1588.

6. *Бобырь, М. В., Н. А. Милостная, и С. Ю. Бельская.* Нечетко-логическая адаптация параметров скользящего окна при подготовке данных для больших языковых моделей // Электронные библиотеки. 2026 т. 29, вып. 4, сс. 1318-37. doi:10.26907/1562-5419-2026-29-4-1318-1337.

7. *Bobyr, M., Milostnaya, N., Belskaya, S.* A Unified Fuzzy Logic Controller for Adaptive Text Generation in Large Language Models // SSRN. 2026. doi: 10.2139/ssrn.6819385.

8. *Klüver J.* A mathematical theory of communication: Meaning, information, and topology // Complexity. 2011. Vol. 16 (3). P. 10–26. https://doi.org/10.1002/cplx.20317

9. *Holtzman A., Buys J., Du L., Forbes M., Choi Y.* The curious case of neural text degeneration // Computation and Language. 2020. arXiv:1904.09751. https://doi.org/10.48550/arXiv.1904.09751

10. *Bobyr M.* Area ratio method for stability analysis of fuzzy logic and neuro-fuzzy systems with bifurcation validation // Information Sciences. 2026. №. 732. P. 122928. https://doi.org/10.1016/j.ins.2025.122928

11. *Бобырь М.В.* Устойчивость в нечетко-логических и нейро-нечетких системах. М: ИНФРА-М, 2026. 271 с.

FUZZY-LOGIC ADAPTATION OF THE DECODING SELECTIVITY COEFFICIENT IN AUTOREGRESSIVE TEXT GENERATION BY LANGUAGE MODELS

M. V. Bobyr[1] [0000-0002-5400-6817], N. A. Milostnaya[2] [0000-0002-3779-9165], S. Y. Belskaia[3] [0009-0000-2013-9972]

[1–3]*Southwest State University, Kursk, Russia*

[1]maxbobyr@gmail.com, [2]nat_mil@mail.ru, [3]s@belskaia.ru

Abstract

In autoregressive text generation using language models (LLM), the next token is selected using a softmax function with a fixed decoding coefficient equal to one, which is the same for all generation steps and all text types. This approach does not take into account the dynamically changing linguistic characteristics of the generated context. For example, a text with high lexical diversity requires high selectivity, meaning the decoding coefficient should be small. A monotone, repetitive context requires a smoother distribution, meaning the decoding coefficient should be large. This article proposes a fuzzy controller for calculating the decoding selectivity coefficient using two linguistic features: the lexical diversity of the generated context and the Shannon entropy of the probability distribution at the previous step. The fuzzy controller is implemented using the Mamdani algorithm with a base of nine fuzzy logic rules and a defuzzifier based on the center-of-gravity method. The effectiveness of the proposed approach lies in the fact that for terminologically rich text, the fuzzy controller reduces the Shannon entropy of the probability distribution by more than 20 times compared to the standard approach.

***Keywords**: Fuzzy inference, Mamdani algorithm, autoregressive generation, Softmax, LLM, selectivity coefficient, TTR, probability entropy, GPT.*

ACKNOWLEDGEMENTS

This work was supported by the Ministry of Science and Higher Education under State Contract no. 075-03-2026-489.

REFERENCES

1. *Vaswani A., Shazeer N., Parmar N. et al.* Attention Is All You Need // Advances in Neural Information Processing Systems. 2017. Vol. 30. P. 6000–6010.

2. *Lee S.E.* Language Models are Unsupervised Multitask Learners (GPT-2) // OpenAI. Seoul National University, 2024. https://cdn.openai.com/better-language-models/language_models_are_unsupervised_multitask_learners.pdf

3. *Rashka S.* Building LLM from Scratch. St. Petersburg: Piter, 2026. 384 p.

4. *Zadeh L.A.* Fuzzy sets // Information and Control. 1965. Vol. 8, No. 3. P. 338–353.

5. *Mamdani E.H.* Application of fuzzy algorithms for control of simple dynamic plant // Proceedings of the Institution of Electrical Engineers. 1974. Vol. 121(12). P. 1585–1588.

6. *Bobyr M.V., Milostnaya N.A., Belskaia S.Yu.* Fuzzy-logic adaptation of sliding window parameters in data preparation for large language models // Electronic libraries. 2026. Vol. 29, Is. 4, pp. 1318-37. doi:10.26907/1562-5419-2026-29-4-1318-1337.

7. *Bobyr, M., Milostnaya, N., Belskaya, S.* A Unified Fuzzy Logic Controller for Adaptive Text Generation in Large Language Models // SSRN. 2026. doi: 10.2139/ssrn.6819385.

8. *Klüver J.* A mathematical theory of communication: Meaning, information, and topology // Complexity. 2011. Vol. 16 (3). P. 10–26. https://doi.org/10.1002/cplx.20317

9. *Holtzman A., Buys J., Du L., Forbes M., Choi Y.* The curious case of neural text degeneration // Computation and Language. 2020. arXiv:1904.09751. https://doi.org/10.48550/arXiv.1904.09751

10. *Bobyr M.* Area ratio method for stability analysis of fuzzy logic and neuro-fuzzy systems with bifurcation validation // Information Sciences. 2026. №. 732. P. 122928. https://doi.org/10.1016/j.ins.2025.122928

11. *Bobyr M.V.* Stability in fuzzy-logic and neuro-fuzzy systems. Moscow: INFRA-M, 2026. 271 p.

СВЕДЕНИЯ ОБ АВТОРАХ

БОБЫРЬ Максим Владимирович – 1978 года рождения, учился в Курском государственном техническом университете (ныне – Юго-Западный государственный университет). В 2012 году защитил диссертацию на соискание доктора технических наук по специальности 05.13.06 «Автоматизация и управление технологическими процессами и производствами». В настоящее время работает профессором на кафедре программной инженерии. Является председателем диссертационного совета по специальности 5.12.4 «Когнитивное моделирование».

Область научных интересов: адаптивные нейро-нечеткие системы вывода, цифровая обработка изображений и распознавание образов, машинное обучение, стереозрение.

Maksim Vladimirovich BOBYR – born in 1978, studied at Kursk State Technical University (now Southwest State University). In 2012, he defended his dissertation for the degree of Doctor of Technical Sciences in the specialty 05.13.06 Automation and Control of Technological Processes and Production. Currently, he works as a professor at the Department of Software Engineering. He is the chairman of the dissertation council for specialty 5.12.4 Cognitive Modeling.

Research interests: adaptive neuro-fuzzy inference systems, digital image processing and pattern recognition, machine learning, stereo vision.

email: maxbobyr@gmail.com

ORCID: 0000-0002-5400-6817

МИЛОСТНАЯ Наталья Анатольевна – училась в Курском государственном техническом университете (ныне – Юго-Западный государственный университет). В 2023 году защитила диссертацию на соискание ученой степени доктора технических наук по специальности 2.3.1. Системный анализ, управление и обработка информации, статистика. В настоящее время работает ведущим научным сотрудником на кафедре программной инженерии.

Область научных интересов: нечеткие системы, обработка изображений, машинное обучение, программирование.

Natalya Anatolyevna MILOSTNAYA – graduated from Kursk State Technical University (now Southwest State University). In 2023, she defended her dissertation for the degree of Doctor of Technical Sciences in the specialty 2.3.1. System Analysis, Control, Information Processing, and Statistics. Currently, she works as a leading researcher at the Department of Software Engineering. Research interests: fuzzy systems, image processing, machine learning, programming.

email: nat_mil@mail.ru

ORCID: 0000-0002-3779-9165

БЕЛЬСКАЯ Светлана Юрьевна – училась в Юго-Западном государственном университете. В настоящее время учится в аспирантуре Юго-Западного государственного университета на кафедре программной инженерии по специальности 5.12.4. Когнитивное моделирование (технические науки). Патентный поверенный РФ, регистрационный номер в реестре 2880.

Область научных интересов: обучающие нечетко-логические системы, LLM.

Svetlana Yurievna BELSKAIA – she studied at Southwestern State University. Currently, he is studying at the graduate school of Southwest State University at the Department of Software Engineering, specializing in 5.12.4. Cognitive Modeling (technical Sciences). Patent Attorney of the Russian Federation, registration number in the register 2880, Research interests: training fuzzy logic systems, LLM.

email: s@belskaia.ru

ORCID: 0009-0000-2013-9972

Материал поступил в редакцию 2 апреля 2026 года