

КОГНИТИВНАЯ АДАПТАЦИЯ ПАРАМЕТРОВ СКОЛЬЗЯЩЕГО ОКНА БОЛЬШИХ ЯЗЫКОВЫХ МОДЕЛЕЙ НА ОСНОВЕ БЫСТРОДЕЙСТВУЮЩЕГО МЕТОДА ОТНОШЕНИЯ ПЛОЩАДЕЙ

Е. О. Брежнева^[0000-0003-0039-3425]

Юго-западный государственный университет, г. Курск, Россия

bregnevaeo@mail.ru

Аннотация

Предложен метод адаптивного управления параметрами скользящего окна больших языковых моделей на основе нечеткого вывода Мамдани с дефаззификацией по быстродействующему методу отношения площадей (MAR-II). Управляющими переменными являются индекс лексического разнообразия (TTR) и средняя длина BPE-токена (ATL), интерпретируемые как измеримые корреляты когнитивной нагрузки текстового фрагмента. Нечеткий регулятор на основе девяти правил вырабатывает шаг скользящего окна (stride) и длину контекста (max_length). Проведено сравнение MAR-II с методом центра тяжести (CoG) на сетке 25 × 25 точек входного пространства (RMSE по stride 99.9 токена, по max_length 147.4 токена). Показано, что MAR-II обеспечивает аддитивность нечеткой модели и устраняет систематические ошибки CoG при времени вычисления одного нечеткого вывода около 35 нс на ПЛИС Xilinx Spartan 3E.

Ключевые слова: когнитивное моделирование, нечеткая логика, метод отношения площадей, скользящее окно, языковая модель, ПЛИС, дефаззификация, нечеткий регулятор Мамдани.

ВВЕДЕНИЕ

Задача формирования скользящего контекстного окна при подаче текста на вход больших языковых моделей (LLM) является важным этапом предобработки данных. Как правило, параметры скользящего окна LLM, такие как шаг (stride) и длина (max_length), задаются фиксированными значениями, не учиты-

вающими лингвистическую неоднородность входного корпуса. Однако семантически насыщенный научный текст и простой повторяющийся текст требуют принципиально различных режимов дискретизации.

Одним из эффективных инструментов адаптивного управления являются нечетко-логические системы. Для дефаззификации в таких системах традиционно применяют метод центра тяжести (CoG, Center of Gravity) [1]. Однако CoG обладает рядом известных недостатков, к которым относятся сужение диапазона дефаззификации, при котором невозможно получить минимальное или максимальное значение суппорта выходной переменной, а также нечувствительность к изменениям входных переменных в случае, когда несколько правил имеют одинаковые заключения [1, 2].

В качестве альтернативы в работах [2, 3] предложены метод отношения площадей (MAR – Method of Areas' Ratio) и его две быстродействующие модификации, включая MAR-II, используемый в настоящем исследовании. Принципиальное преимущество MAR-II состоит в том, что он обеспечивает аддитивность нечеткой системы, а также позволяет получить на выходе любое значение из полного суппорта выходной переменной. Сравнение статистических оценок моделей дефаззификации на основе RMSE и MAPE показало преимущество MAR-II по сравнению с CoG [3]. Реализация MAR-II на ПЛИС Xilinx Spartan 3E в параллельно-конвейерной архитектуре обеспечивает время вычисления одного нечеткого вывода около 35 нс при тактовой частоте 100 МГц [3, 7], что на порядки превосходит программные реализации и делает MAR-II перспективным для встроенных систем реального времени.

Настоящее исследование развивает предшествующую работу авторов [4] в следующих направлениях. Во-первых, при выборе размера скользящего окна вместо классического дефаззификатора CoG использован MAR-II. Во-вторых, введена интерпретация входных переменных в рамках концепции когнитивного моделирования, с использованием двух входных переменных: 1) TTR (Type-Token Ratio – отношение числа уникальных словоформ к общему числу словоформ во фрагменте); 2) ATL (Average Token Length – средняя длина BPE-токена в символах), которые трактуются как измеримые корреляты когнитивной нагрузки [5, 6]. При этом нечеткий регулятор применен как интеллектуальный агент, адаптирующий глубину обработки к семантической плотности текста.

В-третьих, выполнено количественное сравнение CoG и MAR-II на систематической сетке входного пространства.

ПОСТАНОВКА ЗАДАЧИ И КОГНИТИВНАЯ ИНТЕРПРЕТАЦИЯ

Пусть входной текст разбит на фрагменты скользящим окном со следующими параметрами: размер шага $\text{stride} \in [1, 256]$ токенов и длина контекста $\text{max_length} \in [4, 512]$ токенов. Требуется построить систему автоматической настройки этих параметров в зависимости от лингвистических характеристик текущего фрагмента.

В качестве входных переменных нечеткого регулятора приняты следующие переменные и их обозначения. Переменная x_1 обозначает Type-Token Ratio (TTR) $\in [0, 1]$, характеризующий лексическое разнообразие фрагмента, а переменная x_2 – среднюю длину BPE-токена (ATL) $\in [1, 8]$. Высокие значения TTR и ATL соответствуют семантически насыщенному тексту (научный стиль); низкие – повторяющемуся тексту.

Приведем пример вычисления переменных. Пусть фрагмент текста содержит 20 токенов, 17 из которых уникальные. Тогда $\text{TTR} = 17/20 = 0.85$. Если суммарное число символов во всех 20 токенах равно 124, то $\text{ATL} = 124/20 = 6.2$ символа/токен. Данная пара значений ($\text{TTR} = 0.85$, $\text{ATL} = 6.2$) соответствует насыщенному научному тексту и использована в дальнейшем как первый характерный сценарий работы регулятора. Например, фрагмент простого повторяющегося текста из 20 токенов, среди которых лишь 4 уникальных, и средняя длина токена 2.1 символа дают $\text{TTR} = 0.20$, $\text{ATL} = 2.1$.

В рамках концепции когнитивного моделирования скользящее окно LLM интерпретируется как аналог рабочей памяти когнитивного агента [5, 6]. Высокий TTR и длинные токены сигнализируют о необходимости узкого окна с малым шагом. Простой текст допускает широкий шаг и меньший контекст. Предложенный нечеткий регулятор формализует это правило адаптации.

Для количественной характеристики режима работы введен индекс когнитивной плотности обработки $K = \text{max_length}/\text{stride}$. При $K \gg 1$ (большое перекрытие окон) агент работает в режиме глубокого анализа; при $K \approx 1$ имеем режим быстрого сканирования. Например, для научного текста ($\text{TTR} = 0.85$, $\text{ATL} = 6.2$) нечеткий регулятор выдает $\text{stride} = 25$ токенов и $\text{max_length} = 462$ токена, значит,

$K = 462/25 \approx 18.5$. Следовательно, имеем режим плотного перекрытия окон с глубоким анализом каждого фрагмента. Для простого текста ($TTR = 0.20$, $ATL = 2.1$) регулятор выдает $stride = 231$ и $max_length = 52$, получаем $K = 52/231 \approx 0.2$, и это режим быстрого сканирования без перекрытия.

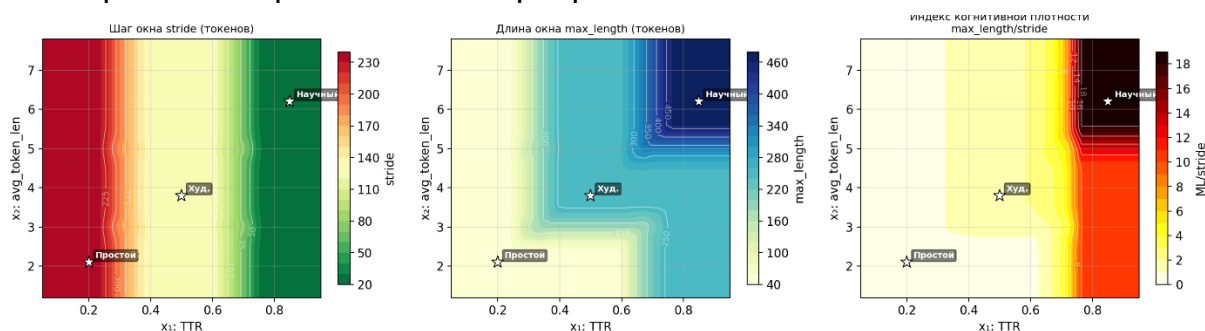


Рис. 1. Карта адаптации параметров скользящего окна в пространстве лингвистических характеристик текста.

Рис. 1 наглядно демонстрирует три характерные зоны адаптации. В левой верхней зоне (низкий TTR, высокий ATL) индекс K принимает промежуточные значения порядка 2–4. Таким образом, длинные токены при малом разнообразии соответствуют технической документации с повторяющейся терминологией. Центральная зона (средние TTR и ATL) соответствует художественному тексту с $K \approx 2$, где регулятор устанавливает сбалансированный режим $stride = 128$, $max_length = 256$. Правая верхняя зона (высокий TTR, высокий ATL) – это область научного текста с максимальным $K \approx 18$, где требуется наиболее детальная обработка. Таким образом, карта на рис. 1 представляет собой визуализацию стратегии адаптивного внимания нечеткого агента в полном пространстве входных переменных.

СТРУКТУРА НЕЧЕТКОГО РЕГУЛЯТОРА

Функции принадлежности входных переменных

Для переменной x_1 (TTR) определены три лингвистических термина с трапециевидными и треугольными функциями принадлежности:

- «низкий» (Low): $\text{trapmf}(x; 0.0; 0.0; 0.20; 0.40)$;
- «средний» (Medium): $\text{trimf}(x; 0.25; 0.50; 0.75)$;
- «высокий» (High): $\text{trapmf}(x; 0.60; 0.80; 1.00; 1.00)$.

Для переменной x_2 (ATL) аналогично определены термы:

- «короткий» (Short): $\text{trapmf}(x; 1.0; 1.0; 2.0; 3.5)$;
- «средний» (Medium): $\text{trimf}(x; 2.5; 4.0; 5.5)$;
- «длинный» (Long): $\text{trapmf}(x; 4.5; 6.0; 8.0; 8.0)$.

Для выходных переменных stride и max_length введены по три терма (Small/Medium/Large и Short/Medium/Long соответственно) с трапециевидными и треугольными функциями принадлежности, охватывающими полный суппорт каждой переменной.

База нечетких правил

База знаний построена на основе матрицы 3×3 (TTR $\times$ ATL) и содержит 9 правил вида «ЕСЛИ x_1 есть A И x_2 есть B, ТО stride есть C И max_length есть D» (табл. 1). Правила сформированы, исходя из следующей логики: высокое лексическое разнообразие и длинные токены $\rightarrow$ малый шаг и длинный контекст (глубокая обработка); низкое разнообразие и короткие токены $\rightarrow$ большой шаг и короткий контекст (быстрое сканирование).

Табл. 1. База нечетких правил нечеткого регулятора.

№	TTR	ATL	stride	max_length	Когнитивный режим
R1	High	Long	Small	Long	Глубокий анализ (научный текст)
R2	High	Medium	Small	Medium	Аналитический
R3	High	Short	Small	Medium	Аналитический
R4	Medium	Long	Medium	Medium	Сбалансированный
R5	Medium	Medium	Medium	Medium	Сбалансированный
R6	Medium	Short	Medium	Short	Сбалансированный
R7	Low	Long	Large	Short	Быстрое сканирование
R8	Low	Medium	Large	Short	Быстрое сканирование

R9	Low	Short	Large	Short	Быстрое сканирование (простой текст)
----	-----	-------	-------	-------	--------------------------------------

Структура базы правил отражает монотонную зависимость параметров скользящего окна от лингвистических характеристик текста. При переходе от высокого TTR к низкому терм выходной переменной *stride* последовательно изменяется с Small на Large, а терм *max_length* – с Long на Short, что соответствует переходу от режима глубокого анализа к режиму быстрого сканирования. Влияние переменной ATL носит уточняющий характер, и при фиксированном TTR увеличение средней длины токена смещает *max_length* в сторону более длинного контекста (правила R1, R4, R7), поскольку длинные токены несут большую семантическую нагрузку и требуют более широкого контекста для корректной интерпретации.

Следует отметить, что три правила (R3, R5, R7) формируют «диагональ» матрицы, в которой оба входных термина имеют одинаковый уровень (High–Short, Medium–Medium, Low–Long). Эти правила активируются при согласованном изменении обеих переменных и задают опорные точки поверхности отклика регулятора. Правила R1 и R9, соответствующие противоположным углам матрицы 3×3, определяют крайние режимы работы системы: максимальную ($K \approx 18.5$) и минимальную когнитивную плотность ($K \approx 0.2$) соответственно.

МЕТОД ДЕФАЗЗИФИКАЦИИ MAR-II

Недостатки метода CoG и обоснование выбора MAR-II

Метод CoG вычисляет выходное значение как взвешенный центр масс агрегированной функции принадлежности, что влечет два систематических недостатка [1, 2]. Во-первых, это сужение диапазона дефаззификации, т. е. метод центра тяжести CoG никогда не достигнет точных граничных значений суппорта выходной переменной, поскольку агрегированная функция имеет конечную ненулевую ширину на нижней и верхней границах. Во-вторых, это нечувствительность дефаззификатора, если несколько правил имеют одинаковые заключения, активация дополнительных правил не изменит координату центра масс, т. е. нечеткая модель не реагирует на изменение входных переменных [2]. Оба эффекта

нежелательны при управлении параметрами скользящего окна, где требуется монотонная реакция регулятора на изменение TTR и ATL.

Метод MAR-II, предложенный в работе [3], устраняет оба отмеченных недостатка. Его вычислительная схема обеспечивает, во-первых, аддитивность, когда любое изменение степени активации правил приводит к изменению выходного значения; во-вторых, полноту суппорта, т. е. выходное значение может принимать любое значение из диапазона $[y_{\min}, y_{\max}]$. При этом число арифметических операций сокращается примерно в 5 раз по сравнению с полным MAR [3], а реализация на ПЛИС обеспечивает время вычисления около 35 нс.

Алгоритм MAR-II для двухвходовой системы Мамдани

Алгоритм MAR-II реализуется в следующей последовательности шагов.

Шаг 1. Фаззификация. Вычисляются степени принадлежности входных термов: $\mu_{TTR}(\text{low})$, $\mu_{TTR}(\text{medium})$, $\mu_{TTR}(\text{high})$ и $\mu_{ATL}(\text{short})$, $\mu_{ATL}(\text{medium})$, $\mu_{ATL}(\text{long})$.

Шаг 2. Нечеткий вывод (MIN по Мамдани). Для каждого из девяти правил вычисляется сила активации:

$$h_i = \min(\mu_{TTR}(A_i), \mu_{ATL}(B_i)), \quad i = 1, \dots, 9.$$

Шаг 3. Агрегация по выходным термам (MAX). Для каждого выходного терма j определяется результирующая степень принадлежности:

$$u_j = \max\{h_i \mid \text{правило } R_i \text{ имеет заключение } j\}, \quad j = 1, \dots, n.$$

Для stride: $u_1 = \max(h_1, h_2, h_3)$ (Small), $u_2 = \max(h_4, h_5, h_6)$ (Medium), $u_3 = \max(h_7, h_8, h_9)$ (Large). Для max_length аналогично, с перегруппировкой по заключениям правил (табл. 1).

Шаг 4. Дефаззификация MAR-II. По формуле (18) из [3] вычисляется отношение площадей:

$$D = \sum_j u_j / (n \omega_j),$$

где $n = 3$ – число выходных термов, $\omega_j = 1$ – весовой коэффициент. Четкое выходное значение определяется по формуле

$$y_{\text{out}} = D(y_{\max} - y_{\min}) + y_{\min}.$$

Принципиальное отличие MAR-II от CoG состоит в следующем. В методе CoG выходное значение определяется как взвешенный центр масс всей агрегированной функции принадлежности, что требует интегрирования по полному диапазону выходной переменной и операции умножения с накоплением. В MAR-II вместо интегрирования используется отношение суммарной площади усеченных треугольных термов к общей площади выходного нечеткого множества, причем площадь каждого усеченного терма аппроксимируется треугольником с основанием b_1 и высотой u_j . Это исключает вычисление длины верхнего основания трапеции b_2 и сокращает число арифметических операций примерно в 5 раз по сравнению с полным MAR [3]. В результате дефаззификация сводится к суммированию трех значений u_j , делению на константу $n = 3$ (реализуемому как сдвиг вправо на один разряд и вычитание) и одному линейному масштабированию, что делает метод эффективно реализуемым в параллельно-конвейерной архитектуре на ПЛИС без аппаратного делителя.

Отметим, что формула MAR-II обеспечивает два ключевых свойства, отсутствующих у CoG. Во-первых, это аддитивность, когда любое изменение степени активации хотя бы одного правила влечет изменение суммы u_j и, следовательно, выходного значения. Во-вторых, это полнота суппорта выходной переменной: при $u_1 = 1, u_2 = u_3 = 0$ получаем $D = 1/3, y_{out} = y_{min} + (y_{max} - y_{min}) / 3$, а при $u_1 = u_2 = u_3 = 1$ получаем $D = 1, y_{out} = y_{max}$. Таким образом, выходное значение может достигать любой точки диапазона $[y_{min}, y_{max}]$ в зависимости от конфигурации активированных правил.

Сравнительный числовой пример

Рассмотрим три характерных режима (табл. 2). Для научного текста ($TTR = 0.85, ATL = 6.2$) активируется единственное правило R1 с силой $h_1 = 1.0$. Агрегация дает $u_{small} = 1.0$, остальные термы равны нулю. CoG, взвешивая агрегированную функцию принадлежности по всему диапазону терма Small, выдает $stride = 25.4$ токена. MAR-II вычисляет $D = 1/3 = 0.333$ и выдает $stride = 86.0$ токена — значение внутри Small, но без экстремального смещения к нижней границе. Для художественного текста ($TTR = 0.5, ATL = 3.8$) активируется R5 с силой

0.867: CoG совпадает с центром терма Medium (128.0), MAR-II выдает 74.7. Расхождение объясняется тем, что CoG взвешивает всю площадь усеченного треугольника, тогда как MAR-II учитывает лишь уровень усечения.

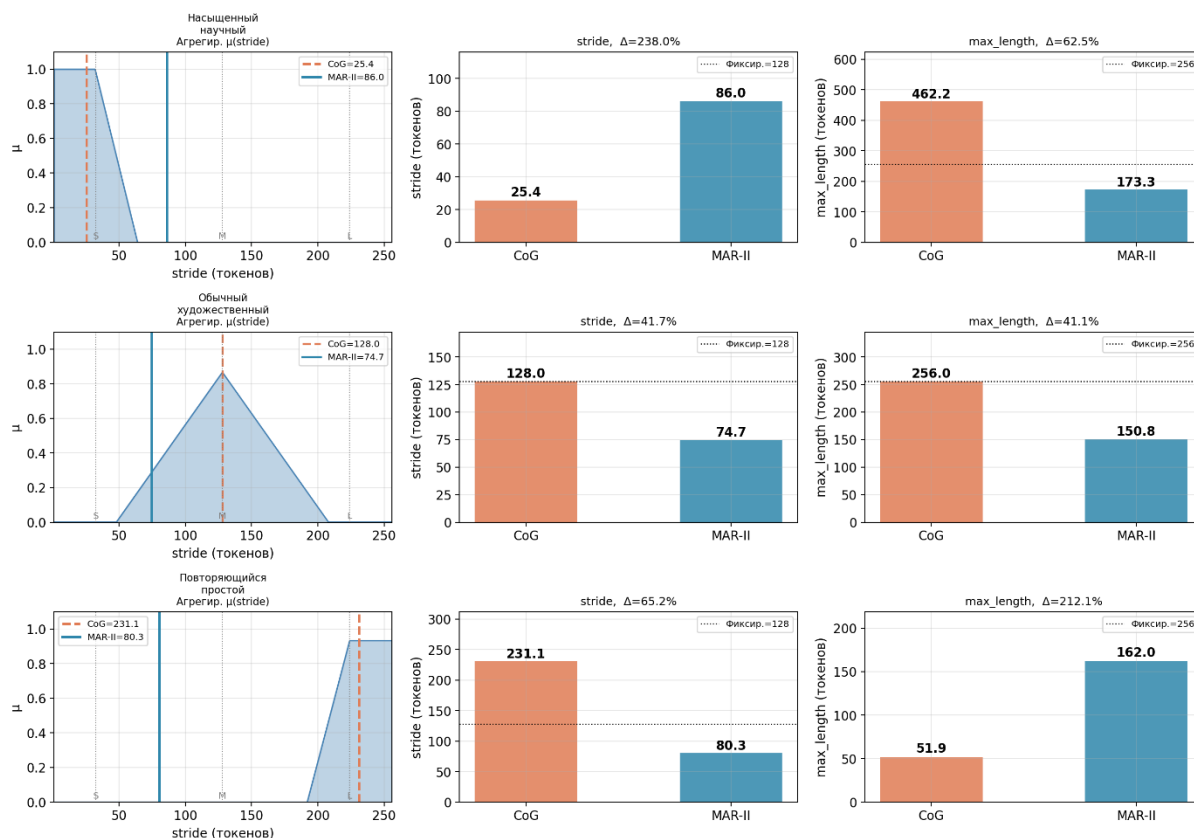


Рис. 2. Сравнение CoG и MAR-II для трех характерных режимов работы нечеткого регулятора.

Табл. 2. Сравнение CoG и MAR-II на трех характерных точках.

Тип текста	TTR	ATL	CoG stride	MAR-II stride	CoG ml	MAR-II ml
Научный	0.85	6.2	25.4	86.0	462.2	173.3
Художественный	0.50	3.8	128.0	74.7	256.0	150.8
Простой	0.20	2.1	231.1	80.3	51.9	162.0

Из табл. 2 видно, что MAR-II систематически смещает значения stride в сторону центра диапазона по сравнению с CoG. Это объясняется равновесным вкла-

дом каждого выходного терма в суммарное отношение площадей. CoG, напротив, при единственном активном правиле смещается к центру масс соответствующего терма, что при крайних (граничных) значениях входных переменных приводит к значительному отклонению от интуитивно ожидаемого выходного значения.

ЭКСПЕРИМЕНТАЛЬНОЕ ИССЛЕДОВАНИЕ

Апробация на текстовом корпусе

Апробация была проведена на корпусе из 25 текстовых фрагментов. Для каждого фрагмента вычислялись TTR (среднее 0.65, диапазон 0.57–0.71) и ATL (среднее 3.5 символа/токен, диапазон 2.5–5.0). Нечеткий регулятор (CoG) выработал *stride* в диапазоне 59 – 128 токенов (среднее 102.1, $\sigma = 24.1$) и *max_length* в диапазоне 93–256 токенов (среднее 218.9, $\sigma = 46.8$). MAR-II выдал *stride* 38–62 токена (среднее 54.0, $\sigma = 6.8$) и *max_length* 69–126 токенов (среднее 104.0, $\sigma = 16.9$). Фиксированные значения *stride* = 128 и *max_length* = 256, использованные в работе [4], не учитывают внутрикорпусную изменчивость и избыточны для фрагментов с низким TTR.

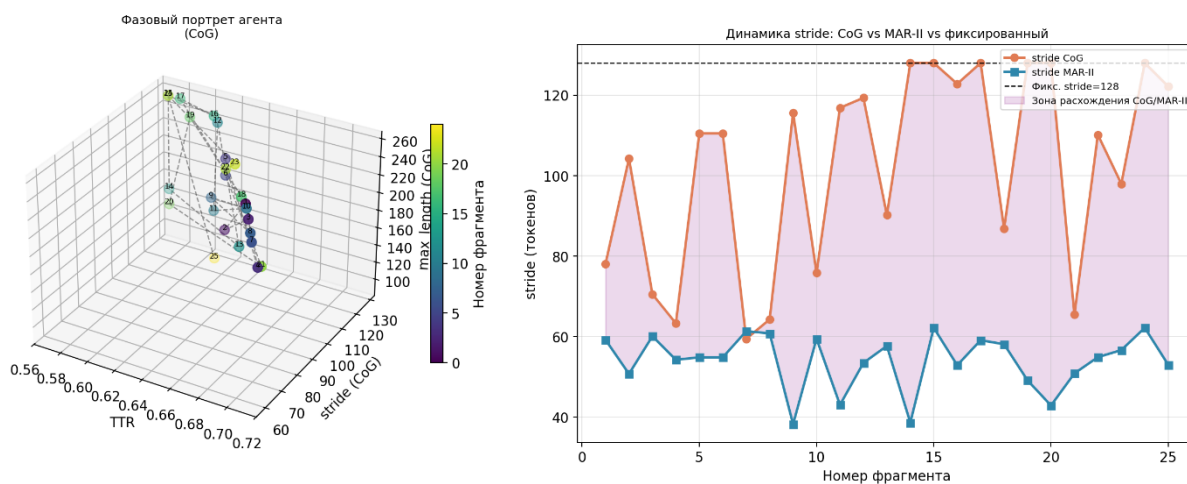


Рис. 3. Траектория интеллектуального агента в пространстве управляющих параметров (25 фрагментов).

Рис. 3 демонстрирует траекторию интеллектуального агента в трехмерном пространстве (TTR, *stride*, *max_length*) при последовательной обработке 25 фрагментов. Траектория не является монотонной, и интеллектуальный агент непрерывно перестраивает оба параметра окна в ответ на флуктуации TTR в диапазоне

0.57–0.71, что подтверждает реактивный характер адаптации. Среднее значение stride CoG (102.1 токена) существенно ниже фиксированного значения 128 токенов, использованного в работе [4], что обеспечивает более плотное покрытие текста и снижает вероятность пропуска смыслового перехода на стыке окон. При этом для 8 из 25 фрагментов (№ 14, 15, 17, 19, 20, 22, 24) CoG выдает максимально возможное значение stride = 128 токенов, соответствующее границе терма Medium, и является характерным примером ошибки сужения диапазона CoG, при которой регулятор «застревает» на границе терма и теряет чувствительность к дальнейшим изменениям входных переменных.

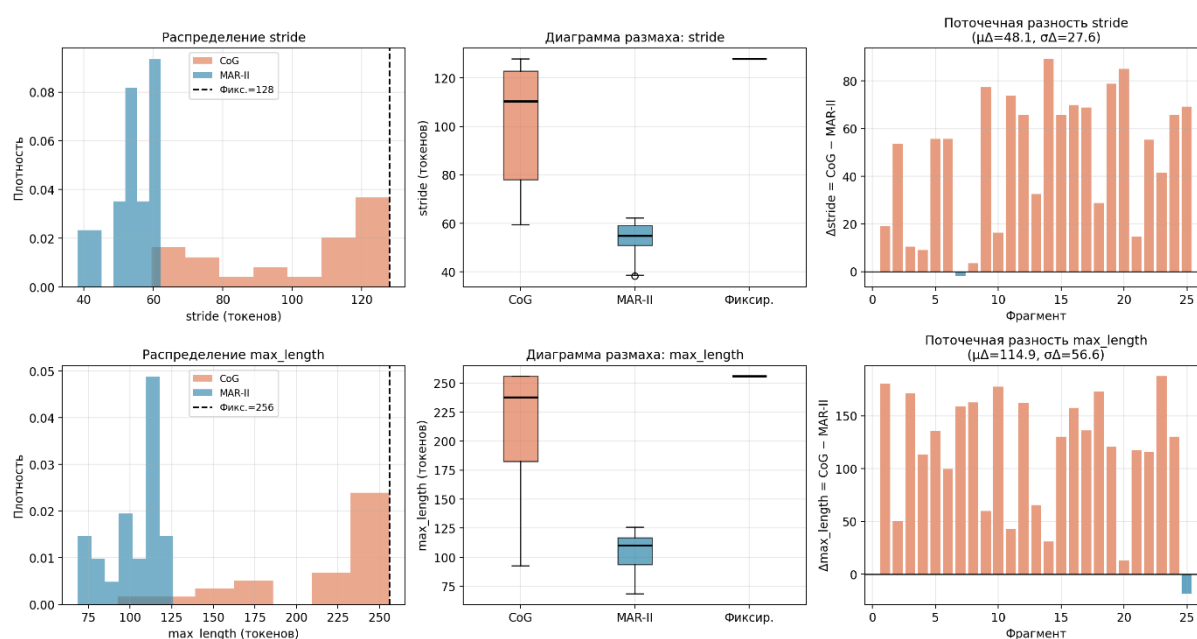


Рис. 4. Статистика параметров нечеткого регулятора при обработке текстового корпуса (25 фрагментов).

Рис. 4 подтверждает это наблюдение статистически. Гистограмма распределения stride для CoG имеет выраженный пик вблизи значения 128 токенов (правая граница диапазона), тогда как распределение MAR-II сосредоточено в диапазоне 38–62 токена со значительно меньшим разбросом ($\sigma = 6.8$ против $\sigma = 24.1$ для CoG). Диаграммы размаха наглядно показывают, что медиана CoG (≈ 110 токенов) вдвое превышает медиану MAR-II (≈ 55 токенов), а межквартильный размах CoG в четыре раза шире. Поточечная разность $\Delta \text{stride} = \text{CoG} - \text{MAR-II}$ положительна для 24 из 25 фрагментов и варьируется от 1.9 до 89.4 токена, что

свидетельствует о систематическом, а не случайном характере расхождения методов. Аналогичная картина наблюдается для `max_length`. Среднее CoG (218.9 токена) вдвое превышает среднее MAR-II (104.0 токена), а разность $\Delta \text{max_length}$ положительна для 24 из 25 фрагментов.

Количественное сравнение CoG и MAR-II на сетке TTR×ATL

Для систематического сравнения были вычислены выходные значения обоих методов на равномерной сетке 25×25 точек

$$\{\text{TTR} \in [0.1, 0.9]\} \times \{\text{ATL} \in [1.5, 7.5]\}.$$

На рис. 5 показано, что поверхность CoG плавная и монотонная, тогда как поверхность MAR-II более равномерная и имеет выраженные зоны постоянного значения (вблизи границ термов), что соответствует свойству полноты суппорта.

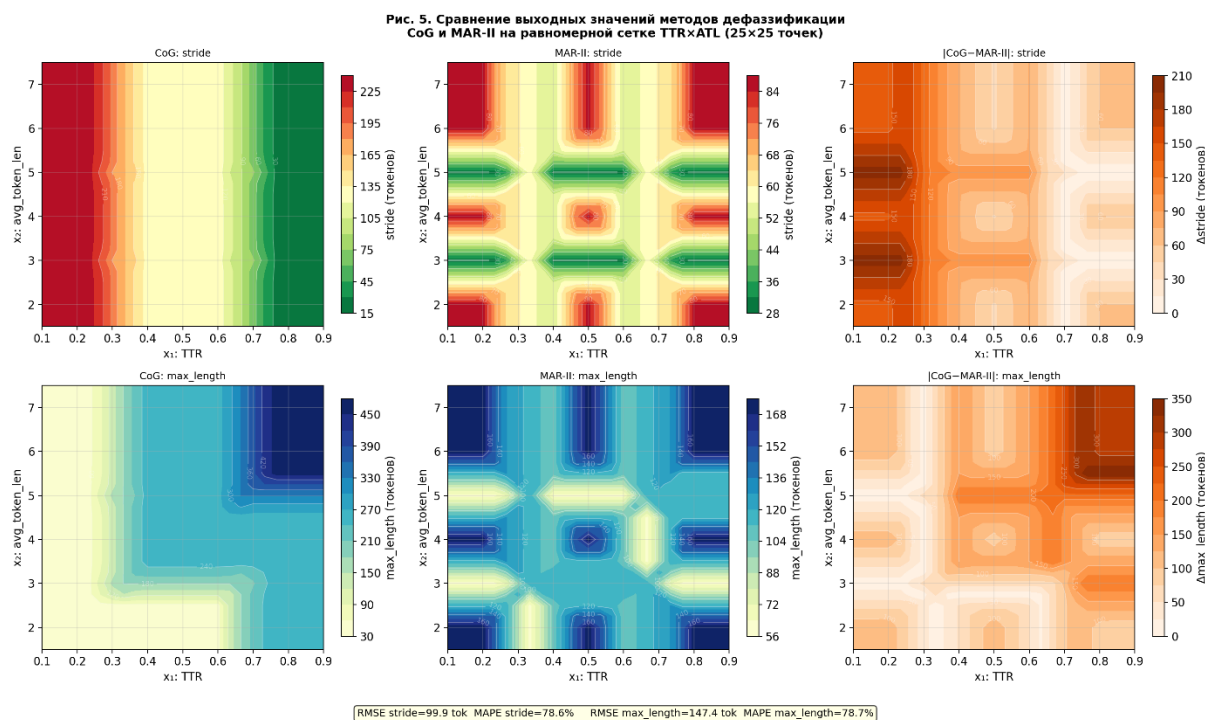


Рис. 5. Сравнение выходных значений CoG и MAR-II на сетке TTR × ATL (25 × 25 точек).

Рис. 5 позволяет проанализировать поведение обоих методов во всем пространстве входных переменных. Поверхность CoG (левый столбец) является плавной и монотонной, где `stride` убывает слева направо по оси TTR от ≈ 230 (низ-

кий TTR) до ≈ 15 токенов (высокий TTR), а `max_length` возрастает в том же направлении от ≈ 30 до ≈ 460 токенов. Влияние ATL на выход CoG выражено значительно слабее, чем влияние TTR, что объясняется доминированием одного активного правила в крайних зонах пространства. Поверхность MAR-II (средний столбец) принципиально отличается по структуре. Она имеет выраженные зоны постоянного значения вблизи границ термов и локальные максимумы в точках пересечения нескольких активных правил, что соответствует свойству аддитивности MAR-II, т. е. одновременная активация нескольких правил с разными заключениями приводит к нелинейному взаимодействию их вкладов через отношение площадей.

Карты абсолютного расхождения $|\text{CoG} - \text{MAR-II}|$ (правый столбец) показывают, что максимальное расхождение по `stride` достигает ≈ 210 токенов в левой части пространства (низкий TTR, любой ATL), где CoG выдает большие значения `stride`, а MAR-II – существенно меньшие значения. По `max_length` максимальное расхождение ≈ 350 токенов наблюдается в правой верхней зоне (высокий TTR, высокий ATL), где CoG выдает значения, близкие к верхней границе диапазона (≈ 460 токенов), тогда как MAR-II — умеренные значения порядка 150–170 токенов. В центральной зоне пространства, соответствующей художественному тексту (TTR ≈ 0.5 , ATL ≈ 4), расхождение минимально для обоих параметров, поскольку здесь доминирует правило R5 (Medium–Medium) с симметричным усечением термина Medium, при котором центр масс CoG и отношение площадей MAR-II дают наиболее близкие результаты.

Табл. 3. Метрики расхождения CoG и MAR-II на сетке 25 × 25.

Параметр	RMSE (токенов)	MAPE (%)
<code>stride</code>	99.9	78.6
<code>max_length</code>	147.4	78.7

Из табл. 3 видно, что значения RMSE и MAPE отражают принципиальное различие в поведении методов, а не погрешность одного из них. Наибольшее расхождение наблюдается в угловых зонах пространства TTR×ATL (зоны активации одного правила), где CoG смещается к центру масс одного термина, тогда как

MAR-II сохраняет вклад незадействованных термов через отношение площадей. По результатам апробации среднее расхождение по stride составило 48.1 токена ($\sigma = 27.6$), по max_length — 114.9 токена ($\sigma = 56.6$). MAR-II выдает меньшие значения stride в 24 из 25 фрагментов, обеспечивая более плотное покрытие текста скользящим окном.

РЕАЛИЗАЦИЯ НЕЧЕТКОГО РЕГУЛЯТОРА НА ПЛИС

Архитектура нечеткого регулятора была реализована в параллельно-конвейерной структуре на базе ПЛИС Xilinx Spartan 3E [3, 7]. Конвейер включает четыре стадии: 1) вычисление степеней принадлежности входных термов; 2) минимизация по правилам (нечеткий вывод); 3) максимизация по выходным термам (агрегация); 4) вычисление отношения площадей MAR-II.

Этап дефаззификации по MAR-II сводится к операциям сложения трех значений u_j и деления на константу $n = 3$, что реализуется как сдвиг и сложение без аппаратного делителя. Это дает существенное преимущество перед CoG, требующим операции умножения и накопления (MAC) для взвешенного центра масс. Время вычисления одного нечеткого вывода составляет около 35 нс при тактовой частоте 100 МГц [3], что делает систему применимой для встроенных устройств предобработки текста в реальном времени.

Функции принадлежности входных и выходных переменных могут быть сохранены в блочной памяти (Block RAM) ПЛИС в виде таблиц поиска (LUT), аналогично подходу, описанному для сигмоидальной функции активации нейрона в [9]. При 12-битной разрядности погрешность вычисления не превышает 0.12% при использовании около 1% ресурсов ПЛИС [9].

ЗАКЛЮЧЕНИЕ

Предложен и исследован метод адаптивного управления параметрами скользящего окна LLM на основе нечеткого регулятора Мамдани с дефаззификацией MAR-II. Получены следующие основные результаты.

1. Разработан нечеткий регулятор с двумя входными (TTR, ATL) и двумя выходными (stride, max_length) переменными, базой из 9 правил и дефаззификацией по методу MAR-II [3].

2. Предложена когнитивная интерпретация: TTR и ATL трактуются как измеримые корреляты когнитивной нагрузки [5, 6]; нечеткий регулятор реализует стратегию адаптивного внимания агента.

3. Проведено количественное сравнение MAR-II и CoG на сетке 25 × 25 точек: RMSE stride = 99.9 токена, RMSE max_length = 147.4 токена. MAR-II обеспечивает аддитивность и полноту суппорта выходной переменной в отличие от CoG.

4. Показана применимость MAR-II для реализации на ПЛИС: время одного нечеткого вывода около 35 нс при 100 МГц, что на порядки превосходит программные реализации [3].

Направления дальнейших исследований будут включать верификацию на реальных LLM-задачах (суммаризация, QA-системы), обучение весовых коэффициентов ω_j нейро-нечеткими методами, а также аппаратную реализацию полного конвейера с измерением ресурсов LUT и FF.

СПИСОК ЛИТЕРАТУРЫ

1. *Leekwijck W.V., Kerre E.E.* Defuzzification: criteria and classification // *Fuzzy Sets and Systems*. 1999. Vol. 108. P. 159–178.

[https://doi.org/10.1016/S0165-0114\(97\)00337-0](https://doi.org/10.1016/S0165-0114(97)00337-0)

2. *Bobyry M.V., Milostnaya N.A., Kulabuhov S.A.* A method of defuzzification based on the approach of areas' ratio // *Applied Soft Computing*. 2017. Vol. 59. P. 19–32. <https://doi.org/10.1016/j.asoc.2017.05.040>

3. *Бобырь М.В., Милостная Н.А., Архипов А.Е., Лунева М.Ю.* Анализ быстросействующих дефаззификаторов в задаче управления роботом-манипулятором // *Вестник компьютерных и информационных технологий*. 2020. Т. 17, № 4. С. 18–28. <https://doi.org/10.14489/vkit.2020.04.pp.018-028>

4. *Бобырь М.В., Милостная Н.А., Бельская С.Ю.* Нечетко-логическая адаптация параметров скользящего окна при подготовке данных для больших языковых моделей llm // *Электронные библиотеки*. 2026. Т. 29, №4. С. 1–21.

5. *Baddeley A.* Working memory // *Science*. 1992. Vol. 255. P. 556–559. <https://doi.org/10.1126/science.1736359>

6. *Miller G.A.* The magical number seven, plus or minus two // *Psychological Review*. 1956. Vol. 63(2). P. 81–97. <https://doi.org/10.1037/h0043158>

7. Bobyr M.V., Yakushev A.S., Dorodnykh A.A. Fuzzy devices for cooling the cutting tool of the CNC machine implemented on FPGA // Measurement. 2020. Vol. 152. P. 107378. <https://doi.org/10.1016/j.measurement.2019.107378>

8. Bobyr M.V., Milostnaya N.A., Bulatnikov V.A. The fuzzy filter based on the method of areas' ratio // Applied Soft Computing. 2022. Vol. 117. P. 108449. <https://doi.org/10.1016/j.asoc.2022.108449>

9. Бондарь О.Г., Брежнева Е.О., Голубев Д.А. Выбор разрядности компонентов нелинейного нейрона при реализации на ПЛИС // Известия Юго-Западного государственного университета. 2025. Т. 29, № 4. С. 70–92. <https://doi.org/10.21869/2223-1560-2025-29-4-70-92>

COGNITIVE ADAPTATION OF SLIDING WINDOW PARAMETERS OF LARGE LANGUAGE MODELS BASED ON A HIGH-SPEED AREA RATIO METHOD

E. O. Brezhneva^[0000-0003-0039-3425]

Southwest state University, Kursk, Russia

bregnevaeo@mail.ru

Abstract

This paper proposes a method for adaptive control of sliding window parameters in large language models (LLMs) based on Mamdani fuzzy inference with defuzzification using the high-speed area ratio method (MAR-II). The input variables are the type-token ratio (TTR) and the average BPE token length (ATL), interpreted as measurable correlates of the cognitive load of a text fragment. A nine-rule fuzzy controller generates the sliding window stride and context length (`max_length`). MAR-II is compared with the center of gravity (CoG) method across a 25×25 grid of input space points (RMSE: 99.9 tokens for stride, 147.4 tokens for `max_length`). It is demonstrated that MAR-II ensures the additivity of the fuzzy model and eliminates the systematic errors of CoG, achieving a single-inference computation time of approximately 35 ns on a Xilinx Spartan 3E FPGA.

Keywords: *cognitive modeling, fuzzy logic, area ratio method, sliding window, language model, FPGA, defuzzification, Mamdani fuzzy controller.*

REFERENCES

1. *Leekwijck W.V., Kerre E.E.* Defuzzification: criteria and classification // *Fuzzy Sets and Systems*. 1999. Vol. 108. P. 159–178.
[https://doi.org/10.1016/S0165-0114\(97\)00337-0](https://doi.org/10.1016/S0165-0114(97)00337-0)
2. *Bobyry M.V., Milostnaya N.A., Kulabuhov S.A.* A method of defuzzification based on the approach of areas' ratio // *Applied Soft Computing*. 2017. Vol. 59. P. 19–32. <https://doi.org/10.1016/j.asoc.2017.05.040>
3. *Bobyry M.V., Milostnaya N.A., Arkhipov A.E., Luneva M.Yu.* Analysis of high-speed defuzzifiers in the problem of robot manipulator control // *Vestnik Komp'yuternykh i Informatsionnykh Tekhnologiy*. 2020. Vol. 17, No. 4. P. 18–28.
<https://doi.org/10.14489/vkit.2020.04.pp.018-028>
4. *Bobyry M.V., Milostnaya N.A., Belskaya S.Yu.* Fuzzy-logic adaptation of sliding window parameters in data preparation for large language models (LLMs) // *Elektronnye Biblioteki [Electronic Libraries]*. 2026. Vol. 29, No. 4. P. 1–21.
5. *Baddeley A.* Working memory // *Science*. 1992. Vol. 255. P. 556–559.
<https://doi.org/10.1126/science.1736359>
6. *Miller G.A.* The magical number seven, plus or minus two // *Psychological Review*. 1956. Vol. 63(2). P. 81–97. <https://doi.org/10.1037/h0043158>
7. *Bobyry M.V., Yakushev A.S., Dorodnykh A.A.* Fuzzy devices for cooling the cutting tool of the CNC machine implemented on FPGA // *Measurement*. 2020. Vol. 152. P. 107378. <https://doi.org/10.1016/j.measurement.2019.107378>
8. *Bobyry M.V., Milostnaya N.A., Bulatnikov V.A.* The fuzzy filter based on the method of areas' ratio // *Applied Soft Computing*. 2022. Vol. 117. P. 108449.
<https://doi.org/10.1016/j.asoc.2022.108449>
9. *Bondar O.G., Brezhneva E.O., Golubev D.A.* Selection of bit width for nonlinear neuron components in FPGA implementation // *Izvestiya Yugo-Zapadnogo Gosudarstvennogo Universiteta [Proceedings of the Southwest State University]*. 2025. Vol. 29, No. 4. P. 70–92. <https://doi.org/10.21869/2223-1560-2025-29-4-70-92>

СВЕДЕНИЯ ОБ АВТОРЕ



БРЕЖНЕВА Екатерина Олеговна – кандидат технических наук, доцент кафедры информационной безопасности Юго-западного государственного университета.

Ekaterina Olegovna BREZHNEVA – Candidate of Technical Sciences, associate professor of the Department of Information Security at Southwest State University.

email: bregnevaeo@mail.ru

ORCID: 0000-0003-0039-3425

Материал поступил в редакцию 2 июня 2026 года