

PROMPTALIGN: АВТОМАТИЧЕСКОЕ ИТЕРАТИВНОЕ ВЫРАВНИВАНИЕ БОЛЬШОЙ ЯЗЫКОВОЙ МОДЕЛИ ПОД ЭКСПЕРТНУЮ РАЗМЕТКУ БЕЗ ДООБУЧЕНИЯ

А. А. Агафонов¹ [0000-0001-6924-6701], А. Р. Хусаинов² [0009-0002-1916-2109],
Н. А. Прокопьев³ [0000-0003-0066-7465]

¹⁻³Казанский (Приволжский) федеральный университет, г. Казань, Россия

¹a.a.agathonov@gmail.com, ²khusainov.akhat@mail.ru,

³nikolai.prokopyev@gmail.com

Аннотация

Большие языковые модели все активнее применяются в социальных и психологических науках, однако их предсказания систематически смещены относительно экспертной разметки (эффект смещения выравнивания, alignment shift), а попытки улучшить инструкции вручную или с помощью примеров приводят к так называемому эффекту «перетягивания одеяла» – деградации миноритарных классов.

В работе предложен PromptAlign – автоматический конвейер оптимизации инструкций, не требующий ни доступа к весам модели, ни участия человека в итеративной коррекции, так как таксономия и пространство признаков задаются экспертом априори. Конвейер извлекает гибридные лингвистические и семантические признаки с помощью большой языковой модели, строит на их основе интерпретируемые L1-логистические регрессии для локализации расхождений с экспертом, выполняет метаанализ ошибок и компилирует новые инструкции, управляя балансом через метрику индекса дисбаланса классов (Blanket Pulling Index, BPI). Эксперимент на корпусе, включающем 541 текст на русском языке, с тремя психологическими классами показал рост macro-F1 на отложенной тестовой выборке с 0.655 до 0.749 и минимальной классовой F1 с 0.449 до 0.656 ($p > 0.99$ по парному бутстрэп-тесту). Метод значительно превосходит базовые уровни — без примеров (zero-shot) и с подачей примеров (few-shot). Анализ вклада компонентов подтвердил критическую роль семантической интерпретации признаков и периодической перекомпоновки инструкций. Итого-

вая инструкция содержит пять правил, отражающих латентные критерии экспертов; в отличие от неинтерпретируемых моделей автоматических оптимизаторов, эти правила доступны для проверки и корректировки специалистом.

Ключевые слова: *большая языковая модель, оптимизация инструкций, выравнивание, интерпретируемость, L1-регуляризация, бутстрэп-тест, классификация текстов, психологическая разметка, суррогатная модель, индекс дисбаланса классов, русскоязычный корпус.*

ВВЕДЕНИЕ

Большие языковые модели (LLM) активно внедряются в исследовательскую практику социальных и поведенческих наук, автоматически выявляя личностные черты, токсичность и агрессию, эмоциональную окраску и коммуникативные стратегии – задачи, еще недавно требовавшие трудоемкой ручной разметки. Однако быстрое распространение этих моделей обнажило методологическую проблему: их предсказания систематически расходятся с эталонными экспертными оценками.

Этот феномен – смещение выравнивания (alignment shift) – порождается неявными предпочтениями, заложенными на стадиях предобучения и обучения с подкреплением на основе отзывов человека (RLHF), и в контекстах высокой содержательной достоверности (психодиагностика, социологический контент-анализ) подрывает надежность автоматической классификации.

Ситуация усложняется, когда дообучение невозможно (например, закрытые коммерческие LLM через API), и единственным инструментом воздействия остается текстовая инструкция (prompt). Ручная итеративная рефлексия эксперта повышает точность, но, как показало предшествующее исследование [1], провоцирует эффект «перетягивания одеяла» (blanket pulling): улучшение целевого миноритарного класса достигается за счет деградации мажоритарных, а сама оптимизация трудоемка, плохо масштабируется и зависит от субъективности аналитика.

Альтернативой служат методы автоматической оптимизации инструкций – Automatic Prompt Engineer (APE) [2], OPRO [3], программный каркас DSPy [4]. Они успешно повышают целевую метрику, но для социальных наук обладают двумя

критическими недостатками: генерируемые инструкции – непрозрачные «черные ящики», не поддающиеся экспертной верификации, а ни один из методов не контролирует баланс точности между классами, что делает многоклассовую оптимизацию рискованной.

Настоящая работа базируется на русскоязычном материале с трехклассовой таксономией «информативное – деструктивное – конструктивное» из [1], где также показано, что предобученные LLM без адаптации смещены в пользу информативного и деструктивного классов за счет конструктивного: граница между информативным текстом (факты, аналитика) и конструктивным (факты плюс эмпатия или решение) остается для модели размытой. Решаемая задача – преодолеть это через автоматизацию цикла коррекции, контроль баланса и экспликацию латентных экспертных критериев в виде правил на естественном языке.

Указанные ограничения ставят следующие исследовательские вопросы.

- Можно ли автоматически итеративно корректировать инструкцию LLM, приближая ее ответы к экспертной разметке, не имея доступа к весам и не привлекая человека к циклу коррекции?
- Можно ли формализовать и автоматически удерживать дисбаланс классов, предотвращая эффект «перетягивания одеяла»?
- Можно ли извлечь из процесса оптимизации интерпретируемые правила на естественном языке, отражающие латентные критерии экспертов?

Ответы на эти вопросы получены с помощью PromptAlign – метода автоматического выравнивания LLM-классификатора под экспертную разметку. Метод реализован в виде конвейера на основе суррогатных интерпретируемых моделей (L1-логистических регрессий), которые локализуют расхождения между экспертом и LLM в пространстве гибридных признаков и транслируют их в правила на естественном языке. Таксономия и признаковое пространство задаются экспертом априори, но цикл коррекции, анализ ошибок и компиляция инструкций автономны; для защиты от деградации миноритарных классов введена метрика BPI (Blanket Pulling Index).

БЛИЗКИЕ ПО ТЕМАТИКЕ РАБОТЫ

Работа опирается на три направления исследований: автоматическую оптимизацию инструкций, выравнивание языковых моделей без доступа к весам и

автоматическую классификацию психологических характеристик текста. Ниже рассмотрены их ключевые результаты и пробелы, которые восполняет PromptAlign.

Автоматическая оптимизация инструкций и контекстное обучение

Ручное конструирование инструкций все чаще уступает автоматическим методам, где инструкция является объектом оптимизации: в APE [2] LLM генерирует и оценивает набор инструкций-кандидатов на размеченном множестве; OPRO [3] формулирует это как численную оптимизацию, где внешняя LLM порождает новые инструкции по значениям целевой функции с предыдущих итераций; DSPy [4] абстрагирует вызовы LLM до декларативных модулей, автоматически подбираемых стохастической оптимизацией.

Альтернативой служит контекстное обучение с подачей примеров (few-shot prompting) [5, 6], где в инструкцию вручную включают репрезентативные примеры каждого класса.

Все перечисленные подходы обладают для социальных наук двумя ограничениями: порождают инструкции-«черные ящики» и оптимизируют одну скалярную метрику без контроля покласового баланса. Как показано далее, простое добавление примеров не спасает от деградации миноритарных классов. PromptAlign наследует идею автоматической генерации, но выстраивает инструкцию целенаправленно, на основе явно извлеченных признаков и правил из систематического анализа расхождений с эталоном.

Выравнивание языковых моделей без дообучения

Задача выравнивания (alignment) ответов модели с человеческими предпочтениями обычно решается изменением параметров: классическая схема RLHF [7, 8] обучает модель наград и затем настраивает LLM с помощью обучения с подкреплением, а более поздние подходы вроде Direct Preference Optimization (DPO) [9] устраняют модель наград, но по-прежнему требуют доступа к весам, т. е. неприменимы, когда LLM доступна только через закрытый API (GPT, Claude, DeepSeek).

В парадигме «выравнивание через инструкцию» тщательно сконструированные инструкции заметно смещают поведение модели: цепочечные рассуждения (chain-of-thought) [10] вызывают сложное логическое поведение за счет

формата инструкции, а ручная итеративная рефлексия эксперта также повышает точность, но, как показано в [1], неизбежно вызывает эффект «перетягивания одеяла».

До сих пор не был реализован систематический метод, который автоматически корректировал бы инструкцию по сравнению с эталонной разметкой и одновременно контролировал бы баланс классов. PromptAlign закрывает этот пробел замкнутым циклом анализа расхождений, генерации правил и компиляции инструкции в чисто инструктивной парадигме с опорой на статистический критерий приемки и метрику BPI.

Классификация психологических и поведенческих характеристик текстов

Значительная часть работ по автоматическому определению психологических и поведенческих особенностей авторов по текстам посвящена выявлению агрессии, токсичности и оскорбительной речи. В работе [11] проведен анализ автоматического обнаружения языка вражды; показано, что модели склонны смешивать оскорбительную лексику с объективно агрессивными высказываниями. В [12] предложена более детальная таксономия, различающая тип и цель оскорбительных сообщений. В области конструктивного общения в [13] представлен корпус комментариев к новостным статьям и продемонстрировано, что отделение конструктивных аргументов от деструктивных требует глубокого семантического анализа. Более широкие личностные модели также извлекаются из текстов: в [14] показано, что лингвистические признаки предсказывают личностные черты с точностью, сравнимой с оценками знакомых людей.

МЕТОДОЛОГИЯ

Метод PromptAlign реализует замкнутый автоматический цикл, приближающий предсказания LLM к экспертной разметке без доступа к ее весам: модель используется для извлечения структурированных лингвистических и семантических характеристик текста, выявления систематических расхождений с экспертом в этом пространстве характеристик и преобразования найденных закономерностей в правила на естественном языке.

Формальная постановка задачи и общая архитектура

Пусть $D = \{(x_i, y_i^*)\}_{i=1}^N$ – выборка текстов с экспертными метками, где $x_i \in X$ – текст, а $y_i^* \in Y = \{1, \dots, K\}$ – один из K классов. Здесь X – пространство (множество) всех возможных текстов на естественном языке, а K – число классов таксономии (в настоящей работе $K = 3$); $Y = \{1, \dots, K\}$ – множество меток классов. Пусть $f_\theta(x, p)$ – большая языковая модель с замороженными параметрами θ , принимающая текст x и текстовую инструкцию $p \in P$. Модель возвращает предсказание $\hat{y} = \operatorname{argmax}_k P_\theta(k | x, p)$.

Целевая метрика – макро-усредненная F1 – имеет вид:

$$\text{macro-F1}(p) = \frac{1}{K} \sum_{k=1}^K \frac{2 TP_k(p)}{2 TP_k(p) + FP_k(p) + FN_k(p)}.$$

Задача – найти инструкцию p^* , максимизирующую макро-F1 без превышения допустимой деградации наихудшего класса:

$$p^* = \operatorname{argmax}_{p \in P} \text{macro-F1}(p) \text{ s.t. } \min_k F1_k(p) \geq \min_k F1_k(p_0) - \epsilon.$$

Здесь p_0 – начальная базовая инструкция, $\epsilon \geq 0$ – разрешенное снижение минимальной классовой F1. Это ограничение противодействует эффекту «перетягивания одеяла» – улучшению одних классов ценой резкого падения других.

Пространство P дискретно и экспоненциально велико, поэтому PromptAlign строит приближение p^* итеративно (общая архитектура конвейера показана на рис. 1), как следующую последовательность шагов.

1. *Классификация*. Текущая инструкция p_t применяется к обучающему и валидационному подмножествам, формируются предсказания $\hat{y}_i^{(t)} = f_\theta(x_i, p_t)$.

2. *Локализация ошибок*. Выделяется множество $E_t = \{i \mid \hat{y}_i^{(t)} \neq y_i^*\}$.

3. *Извлечение признаков*. Вычисляется гибридный признаковый вектор $\phi(x_i)$.

4. *Обучение суррогатных моделей*. Строятся две интерпретируемые L1-логистические регрессии (экспертная и LLM-версия); по разности их коэффициентов определяются признаки, по которым модель расходится с экспертом (формально – ниже).

5. *Метаанализ ошибок.* Стратифицированная выборка из E_t подается LLM с инструкцией сформулировать системные причины ошибок и породить обобщенные правила.

6. *Компиляция инструкции.* Правила и семантическая интерпретация расходящихся признаков встраиваются в инструкцию инкрементально; при накоплении правил делается полная перекомпоновка «с нуля».

7. *Оценка и приемка.* Новая инструкция p_{t+1} оценивается на валидации по парному бутстрэп-тесту и метрике BPI (индекс дисбаланса классов, формализован ниже).

Далее каждый шаг рассмотрен подробно.

Начальная инструкция p_0 содержит только краткие определения трех классов без примеров (см. Приложение А). Классификация выполняется с температурой 0.0 для максимальной воспроизводимости.

Признаковое описание текста $\phi(x)$:

- детерминированные признаки (всего 13) – частотные и статистические показатели без LLM (см. Приложение Б); стабилизирующий «якорь», не зависящий от стохастичности модели;
- LLM-признаки (всего 10) охватывают недоступные правилам семантические аспекты: текст подается в ту же LLM ($t = 0.1$) с запросом JSON-объекта с бинарными, непрерывными и категориальными полями (см. Приложение Б).

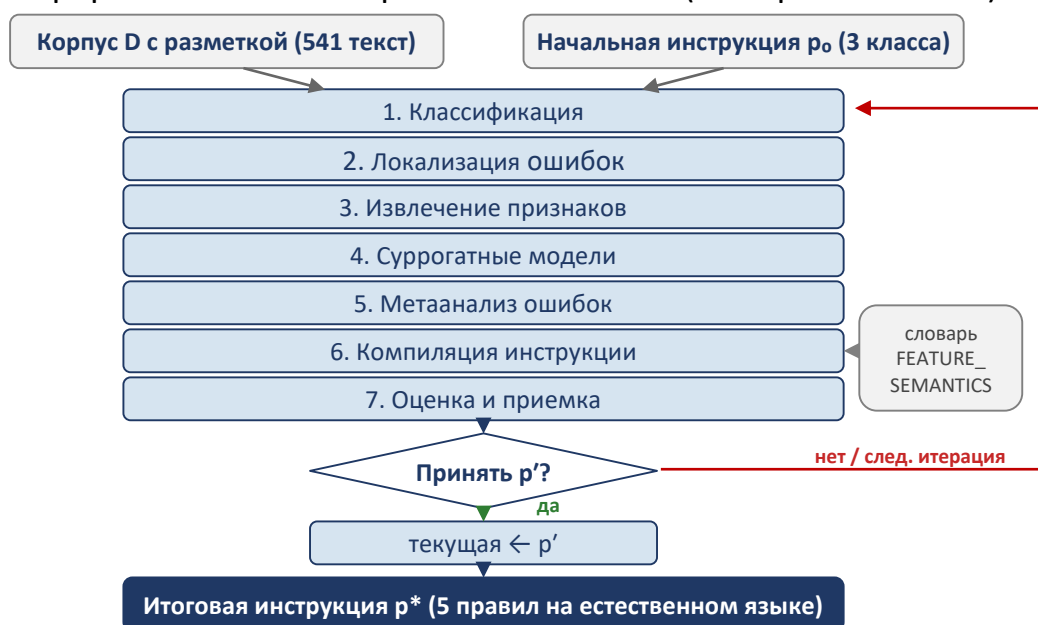


Рис. 1. Архитектура конвейера PromptAlign.

Для повышения устойчивости каждый текст обрабатывается экстрактором дважды ($n_{\text{votes}} = 2$). Бинарные поля усредняются мажоритарным голосованием, непрерывные – средним, категориальный `speech_act` кодируется унитарно (one-hot). Итоговый вектор $\phi(x)$ образуется конкатенацией 13 детерминированных характеристик и преобразованных LLM-признаков.

На матрице признаков X (все числовые поля лежат в отрезке $[0, 1]$) обучаются две мультиномиальные логистические регрессии:

$$g_{\text{exp}}(x) = \operatorname{argmax}_k P_{\text{exp}}(k | x), \quad g_{\text{LLM}}(x) = \operatorname{argmax}_k P_{\text{LLM}}(k | x).$$

Первая модель предсказывает экспертную метку y^* , вторая – метку, которую присвоила LLM ($\hat{y}^{\text{LLM}}$). Обе обучаются с L1-штрафом (lasso) [15], что автоматически обращает в нуль коэффициенты нерелевантных переменных и дает разреженное, легко интерпретируемое решение; при небольшой обучающей выборке такая регуляризация особенно важна, так как подавляет переобучение и выделяет действительно значимые признаки.

Для каждого j -го признака вычисляется величина

$$\delta_j = \frac{1}{K} \sum_{k=1}^K |w_{kj}^{\text{exp}}| - \frac{1}{K} \sum_{k=1}^K |w_{kj}^{\text{LLM}}|,$$

где w_{kj}^{exp} , w_{kj}^{LLM} – веса j -го признака в k -м классе для экспертной и LLM-моделей соответственно. Если $\delta_j > 0$, эксперт использует этот признак интенсивнее, чем LLM; если $\delta_j < 0$, признак избыточно эксплуатируется моделью.

Восемь признаков с наибольшими абсолютными значениями δ_j передаются генератору инструкции через фиксированный словарь `FEATURE_SEMANTICS`, переводящий числовые расхождения в содержательные указания (например, для `has_facts`: «Наличие фактов, цифр и статистики – главный признак Информативного класса»).

Из множества ошибок E_t формируется стратифицированная подвыборка (не более двух примеров на каждый из шести переходов «истинный класс → предсказанный класс», не более 15 текстов всего) – баланс представительности ошибок и затрат на API.

Каждый отобранный текст снабжается метаинструкцией, которая требует от LLM проанализировать причину ошибки и сформулировать обобщенное правило в формате JSON (поля *reasons* и *rule*). Пример: «Если текст содержит критику, аргументацию и предложение решения (даже резкое), то класс Конструктивный, а не Деструктивный». Правила накапливаются в наборе до 20 записей, дубликаты отбрасываются.

Каждая итерация включает инкрементальную компиляцию: текущая инструкция p_t дополняется не более чем четырьмя свежими правилами из метаанализа и семантическими подсказками о признаках с наибольшими δ_j . Генератор (та же LLM) сохраняет исходную структуру, лишь добавляя переданные правила.

Когда получено не менее пяти правил и пройдено две итерации, запускается полная перекомпоновка. Генератор инструкций получает базовый шаблон (p_0), весь набор правил и подсказки, синтезируя новую инструкцию с обобщением правил и устранением противоречий и избыточности. Постобработка гарантирует маркер {text} и фразу «Ответь ОДНИМ СЛОВОМ (Информативный/Деструктивный/Конструктивный)»; при превышении лимита в 1500 символов следует запрос на сокращение.

Статистический критерий принятия инструкции и контроль баланса

Решение о замене текущей лучшей инструкции p_{best} на кандидата p_{t+1} опирается на комбинацию бутстрэп-теста и анализа поклассовых метрик.

Парный бутстрэп-тест. На валидационной выборке вычисляется разность *macro-F1* между новой и лучшей инструкциями. Генерируется 1000 бутстрэп-выборок ресэмплингом с возвращением [16] по парам «текст – истинная метка» с пересчетом той же разности; по эмпирическому распределению оцениваются:

- p_{better} – доля бутстрэп-выборок, где новая инструкция показала более высокую *macro-F1*;
- 95%-й доверительный интервал (CI) для разности.

Новая инструкция принимается, если выполняется хотя бы одно из двух условий:

- 1) $p_{better} \geq 0.75$;

2) прирост макро-F1 превышает 0.015 при снижении минимальной классовой F1 не более чем на 0.05 от p_{best} .

Такой составной критерий выработан эмпирически: он обеспечивает чувствительность к реальным улучшениям, не пропуская при этом ложные срабатывания на малых выборках.

Контроль «перетягивания одеяла». Параллельно вычисляется индекс дисбаланса классов (Blanket Pulling Index, BPI):

$$\text{BPI}_t = \max_k \Delta F_{1,k} - \min_k \Delta F_{1,k}, \quad \Delta F_{1,k} = F_{1,k}(p_{t+1}) - F_{1,k}(p_{\text{best}}).$$

BPI измеряет размах изменений классовой F1: большое значение говорит о том, что рост одних классов получен за счет падения других. Если худший класс деградирует более чем на 0.03, а BPI превышает 0.15, то кандидат отвергается даже при пройденном бутстрэп-тесте.

Все пороговые параметры (0.75; 0.015; 0.05; 0.03; 0.15) зафиксированы для всех экспериментов и подобраны предварительными тестами для консервативной приемки улучшений при ограниченном объеме валидационных данных.

ОРГАНИЗАЦИЯ ЭКСПЕРИМЕНТА

Исходные данные

Был использован тот же корпус из 541 текстов на русском языке, что и в [1], собранный с форумов, из социальных сетей и блогов. Эксперты-психологи разметили каждый текст по трехклассовой таксономии: информативный (нейтральная передача фактов, аналитика, статистика без выраженной авторской позиции и призывов к действию), деструктивный (агрессия, провокации, токсичность, выражение безысходности и эмоционального страдания без попыток решения) и конструктивный (эмпатия, сотрудничество, предложение решений, критика с аргументацией и рекомендациями).

Выборка стратифицированно была разделена на три части ($\text{random_state} = 42$). Тестовый набор (20% от корпуса, $N_{\text{test}} = 109$) зарезервирован исключительно для итоговой оценки. Оставшиеся 432 текста разделены на обучающую ($N_{\text{train}} = 302$, что составляет 70% от оставшихся 432 текстов) и валидационную ($N_{\text{val}} = 130$, что составляет 30% от тех же 432 текстов) выборки. Распределение классов представлено в табл. 1.

Табл. 1. Распределение классов по выборкам.

Выборка	Информативный	Деструктивный	Конструктивный	Всего
Обучающая	101	101	100	302
Валидационная	43	43	44	130
Тестовая	36	36	37	109
Итого	180	180	181	541

Языковая модель и протокол взаимодействия

Все запросы отправлялись через публичный API deepseek-chat (DeepSeek-V3) с температурой 0.0 для классификации и метаанализа (воспроизводимость), 0.1 для признаков (разнообразие при усреднении) и 0.2 для генерации инструкции (баланс креативности и контроля).

Для извлечения признаков каждый текст обрабатывали дважды ($n_{\text{votes}} = 2$) с усреднением результатов (см. Методологию). Длинные тексты (свыше 4000 символов) обрезали до последнего полного предложения. Вызовы соблюдали ограничения API (28 запросов/мин, 12 000 токенов/мин) с автоматическим ожиданием при превышении лимитов.

Параметры PromptAlign

Конфигурация конвейера зафиксирована заранее и не изменялась в ходе итераций (табл. 2); значения подобраны предварительными тестами на подмножестве данных без оптимизации под финальный результат, чтобы избежать переобучения.

Табл. 2. Параметры конвейера.

Параметр	Значение
Максимальное число итераций	8
Бутстрэп-повторов	1000
Порог p_{better}	≥ 0.75
Альтернативный порог прироста macro-F1	> 0.015
Допустимое снижение min-F1	≤ 0.05
Порог BPI для отклонения	> 0.15
Периодичность перекомпоновки	Каждые 2 итерации

Максимальное число правил в наборе	20
Максимальная длина инструкции	1500 символов
С для логистической регрессии	0.5
Решатель логистической регрессии	liblinear
Автоматическое расширение признаков	Выключено

Вычислительные ресурсы и воспроизводимость

Конвейер реализован на Python 3.10 (scikit-learn 1.3, pandas 2.0, numpy 1.24, openai 1.0) в среде Google Colab (Intel Xeon, 12 ГБ ОЗУ). Для полной воспроизводимости все ответы LLM кэшировались в файлах JSON и восстанавливались при повторном запуске, что устраняло влияние случайности и позволяло продолжать прерванные вычисления без потерь.

Полный цикл из восьми итераций занял около 20 ч (почти все время – ожидание ответов API) и около 2600 вызовов API; обучение регрессий и бутстрэп-тесты заняли менее минуты. Стоимость по тарифам DeepSeek (июнь 2025 г.) не превысила 1–2 долларов США, т. е. метод доступен даже при минимальном бюджете (для сравнения: few-shot – 240 вызовов, без семантического перевода – 130 USD). Код, данные и результаты доступны в открытом репозитории (<https://github.com/AlexanderAgathonov/PromptAlign>).

РЕЗУЛЬТАТЫ

Основной результат

Полный цикл из восьми итераций дал два статистически значимых улучшения; дальнейшие модификации конвейера устойчиво отклонял.

Динамика на валидационной выборке показана в табл. 3. Базовая инструкция p_0 (лишь краткие определения классов) показала $\text{macro-F1} = 0.373$ и $\text{min-F1} = 0.219$ – наихудшим был конструктивный класс. Первая итерация (инкрементальная генерация с семантическими подсказками) подняла macro-F1 сразу на +0.265. На второй итерации сработала полная перекомпоновка на основе 22 накопленных правил, добавив еще +0.139 по macro-F1 и доведя min-F1 до 0.723. Итерации 3–8 не проходили пороги бутстрэп-теста и BPI, поэтому были отвергнуты.

Разрыв между базовыми значениями на валидации (0.373) и тесте (0.655) объясняется тем, что валидационная выборка при стратифицированном разбиении случайно оказалась сложнее для исходной инструкции; поскольку сравнения ведутся на одних и тех же разбиениях, относительный прирост остается корректным.

На отложенной тестовой выборке (табл. 4) PromptAlign статистически значимо превзошел базовый уровень (zero-shot) по всем ключевым метрикам: macro-F1 вырос с 0.655 до 0.749, min-F1 – с 0.449 до 0.656. Парный бутстрэп-тест (1000 повторов) дает 95%-ный доверительный интервал для разности macro-F1 $[+0.014; +0.174]$ и $p_{\text{better}} = 0.992$.

Табл. 3. Динамика качества на валидационной выборке (130 текстов).

Итерация	Действие	macro-F1	min-F1	BPI	Статус
0 (базовый)	–	0.373	0.219	–	–
1	Инкрементальная генерация	0.638	0.608	0.195	Принят
2	Полная перекомпоновка	0.777	0.723	0.051	Принят
3–8	Инкрементальная перекомпоновка	≤ 0.703	≤ 0.609	–	Отклонен

Примечание: в строке «3–8» приведены максимальные значения метрик среди отклоненных итераций.

Табл. 4. Сравнение на тестовой выборке (109 текстов).

Метрика	Базовый уровень (zero-shot)	PromptAlign
macro-F1	0.655	0.749
min-F1	0.449	0.656
F1 Информативный	0.706	0.756
F1 Деструктивный	0.810	0.833
F1 Конструктивный	0.449	0.656
Коэффициент каппа Коэна (κ)	0.531	0.628
BPI (относительно базового уровня)	–	0.183

Основная неопределенность сохраняется между конструктивным и информативным классами, но ее масштаб существенно уменьшился. Базовая инструкция ошибочно относила 17 из 36 конструктивных текстов к информативным; PromptAlign сократил число таких ошибок до 11. Интегральный показатель дисбаланса BPI = 0.183 подтверждает, что прирост min-F1 на +0.207 достигнут без сильной деградации других классов: величина F1 информативного класса выросла на 0.050, деструктивного – на 0.023.

Сравнение с базовым уровнем «несколько примеров» (few-shot)

PromptAlign сравнивался с базовым уровнем few-shot prompting – инструкцией с вручную добавленными тремя репрезентативными примерами каждого класса из обучающей выборки. Примеры были подобраны так, чтобы выразительно проиллюстрировать маркеры классов и не превысить разумные ограничения контекста (см. Приложение В; полные тексты – в репозитории).

Результаты на тестовой выборке представлены в табл. 5. Метод с подачей примеров дает лишь незначительный прирост над методом без примеров (macro-F1 на +0.017, min-F1 на +0.042) и одновременно ухудшает распознавание деструктивного класса (F1 с 0.810 до 0.782). Это прямое проявление «перетягивания одеяла»: акцент на конструктивных и информативных примерах смещает вероятности модели в ущерб деструктивному классу.

PromptAlign значительно превосходит метод с подачей примеров (few-shot) по всем метрикам (macro-F1 выше на 0.077, min-F1 – на 0.165), причем наибольший прирост приходится на конструктивный класс (+0.165) благодаря явному контролю баланса через BPI и суррогатные модели.

Табл. 5. Сравнение методов на тестовой выборке (109 текстов).

Метод	macro-F1	min-F1	F1 Информ.	F1 Дестр.	F1 Констр.
Метод без примеров (базовый)	0.655	0.449	0.706	0.810	0.449
Метод с подачей примеров (3 примера/класс)	0.672	0.491	0.744	0.782	0.491
PromptAlign	0.749	0.656	0.756	0.833	0.656

Анализ вклада компонентов

Чтобы оценить роль ключевых архитектурных решений, был проведен анализ вклада компонентов (ablation study). Полный перезапуск конвейера в различных конфигурациях крайне ресурсоемок, поэтому результаты основного эксперимента комбинировались с контролируруемыми локальными тестами. Конфигурация «Без семантического перевода» получена работой генератора инструкций на данных первой итерации, но с передачей ему сырых чисел δ_j вместо словаря FEATURE_SEMANTICS. Конфигурация «Без перекомпоновки» соответствует состоянию после первой (инкрементальной) итерации. Конфигурация «Жесткий критерий» моделирует применение жесткого порога $p_{\text{better}} \geq 0.85$ вместо составного мягкого критерия.

Результаты на валидационной выборке представлены в табл. 6.

Табл. 6. Анализ вклада компонентов (валидационная выборка, 130 текстов).

Конфигурация	macro-F1	min-F1	Δ macro-F1
Полный метод (итерация 2)	0.777	0.723	–
Без семантического перевода	0.612	0.559	–0.165
Без перекомпоновки (только итерация 1)	0.638	0.608	–0.139
Жесткий критерий ($p \geq 0.85$)	0.777	–	0.000

Анализ вклада компонентов (табл. 6) позволяет сформулировать три ключевых наблюдения.

1. *Семантический перевод признаков.* Без словаря FEATURE_SEMANTICS генератор инструкций не получает из сырых числовых расхождений весов достаточного контекста для синтеза осмысленных правил (табл. 6) – лингвистическая трансляция статистики оказывается критичной.

2. *Периодическая перекомпоновка.* Инкрементальная компиляция дает прирост, но без обобщения накопленных правил качество стабилизируется (табл. 6); полная перекомпоновка устраняет противоречия и избыточность в разросшейся инструкции.

3. *Мягкий критерий принятия.* В данном эксперименте более жесткий порог не изменил бы итогового результата, но критерий служит защитой от ложных срабатываний на более шумных или малых выборках.

ОБСУЖДЕНИЕ

Интерпретируемость и экспликация экспертных критериев

Одним из главных требований к автоматическим системам в социальных и поведенческих науках является прозрачность решений [17, 18]. В отличие от APE или OPRO, порождающих «черные ящики», итоговая инструкция PromptAlign содержит пять обобщенных правил (см. Приложение А), которые напрямую отражают критерии экспертов-психологов, не сформулированные ими явно до эксперимента, и доступны для независимой проверки специалистом.

Например, правило, предписывающее относить тексты с фактами и аналитикой без предложения решений к информативному классу, устраняет проблему из [1]: модель склонна была принимать любую выраженную авторскую позицию за признак конструктивного поведения. Таким образом, конвейер не просто повышает метрики, но и служит инструментом экспликации неявного экспертного знания и уточнения таксономий.

Анализ оставшихся ошибок: семантические границы и ирония

Несмотря на значительный прогресс, смешение конструктивных и информативных классов остается основным источником ошибок (11 из 36 конструктивных текстов отнесены к информативным): такие тексты содержат развернутый анализ, аргументацию и выводы, но лишены явного призыва к действию или предложения решения – граница между «глубокой аналитикой с авторской позицией» и «конструктивным разбором причин и следствий» оказывается тонкой и для эксперта, и для модели.

Более сложную проблему создает культурная специфика русскоязычной иронии: в примере из Приложения В текст формально изобилует позитивной лексикой («толковая», «верно», «грамотно», «красиво»), но его реальный смысл состоит в едкой насмешке и скрытой агрессии. Итоговая инструкция указывает, что сарказм может маскировать деструктив, но не дает формальных маркеров

для его распознавания. Эта проблема хорошо известна в компьютерной лингвистике [19] и особенно остра для русского языка.

Сравнение с ручной рефлексией и обучением с подачей примеров

Сопоставление с ручной итеративной рефлексией [1] демонстрирует преимущество автоматического контроля баланса: ручная оптимизация в [1] достигла пиковых F1 (0.972 информативный, 0.967 деструктивный), но min-F1 не превысила 0.619 из-за «перетягивания одеяла», тогда как PromptAlign обеспечил min-F1 = 0.656 при одновременном росте F1 информативного (0.756) и деструктивного (0.833) классов, т. е. уступает в пиковых значениях, но дает более сбалансированное и устойчивое решение, что для прикладных социальных исследований часто предпочтительнее, причем весь цикл коррекции выполняется без участия эксперта, в отличие от ручной рефлексии.

Сравнение с методом с подачей примеров (табл. 5) подтверждает, что простое добавление примеров не устраняет смещение выравнивания и не спасает от деградации деструктивного класса – явное извлечение признаков, локализация расхождений и генерация правил заметно эффективнее неявного обучения на примерах.

Ограничения метода и эксперимента

Перечислим основные ограничения.

1. Объем выборки, статистическая мощность и локальная сходимость. При выборке из 541 текста доверительные интервалы бутстрэп-теста остаются широкими, а правила метаанализа могут быть переобучены на формулировки обучающей выборки и не обобщаться на иные жанры. Кроме того, конвейер гарантирует улучшение лишь в окрестности стартовой инструкции p_0 ; сходимость к глобальному оптимуму в пространстве всех текстовых инструкций не гарантируется.

2. Граница применимости: экспертное проектирование таксономии и словаря. Метод не автоматизирует проектирование таксономии и признакового пространства, это требует участия эксперта в каждой новой области (определить классы, признаки, словарь FEATURE_SEMANTICS), хотя сама архитектура от предметной области не зависит (см. «Перспективы»). Автоматизирован только цикл поиска правил после того, как классы и признаки заданы.

3. Ограничение длины инструкции и зависимость от качества LLM-признаков. Лимит в 1500 символов (экономия, компактность) ограничивает сложность и число представимых правил; отдельно, плохое распознавание иронии, эмпатии или скрытой агрессии зашумляет суррогатные признаки, а пороги BPI (0.15 и 0.03), подобранные эмпирически, при смещении могут изменить набор принимаемых инструкций, хотя общая логика защиты от дисбаланса устойчива.

Перспективы применения и дальнейшие исследования

Архитектура PromptAlign универсальна по отношению к предметной области, однако ее перенос в новую область по-прежнему требует априорного участия эксперта (см. разд. «Ограничения метода и эксперимента»): замены описаний классов, адаптации набора признаков и предоставления размеченной выборки. Потенциальные сферы применения включают медицинскую классификацию по срочности (триаж), модерацию пользовательского контента, анализ юридических документов и выявление маркеров психологического неблагополучия.

Среди направлений дальнейших исследований – автоматическая ранняя остановка по динамике прироста метрик вместо фиксированного лимита итераций; ансамблирование нескольких лучших инструкций для устойчивости предсказаний; интеграция контекстных векторных представлений для распознавания иронии и скрытой агрессии; масштабирование и валидация метода на более крупных мультязычных наборах данных.

ЗАКЛЮЧЕНИЕ

Предложен и экспериментально исследован PromptAlign – автоматический конвейер, выравнивающий предсказания LLM под экспертную разметку без дообучения весов и решающий две ключевые проблемы: систематическое смещение и «перетягивание одеяла» – деградацию миноритарных классов при ручной или контекстной оптимизации инструкций.

PromptAlign опирается на гибридные признаки, интерпретируемые L1-логистические регрессии для выявления расхождений с экспертом и метаанализ ошибок, порождающий правила на естественном языке. Эксперимент на корпусе из 541 текста на русском языке (три класса) показал: уже за две итерации конвейер формирует инструкцию, статистически значимо превосходящую и

zero-shot-, и few-shot-подходы, а также ручную итеративную рефлексия эксперта. Величина Macro-F1 на отложенной тестовой выборке выросла с 0.655 до 0.749, минимальная классовая F1 – с 0.449 до 0.656 ($p > 0.99$ по парному бутстрэп-тесту). Метрика BPI удержала «перетягивание» на низком уровне (0.183), позволив усилить конструктивный класс без потерь для остальных.

Итоговая инструкция содержит пять прозрачных правил: в отличие от «черных ящиков», порождаемых многими оптимизаторами, они не просто повышают метрики, но эксплицируют латентные критерии экспертов и доступны специалисту для независимой проверки и коррекции.

Предложенный конвейер универсален по отношению к предметной области и применим в любых задачах автоматической классификации текстов, где доступна экспертная разметка и важна прозрачность принимаемых решений.

Благодарности

Исследование выполнено за счет гранта Российского научного фонда № 24-18-01125 «Психологическая модель поведения субъекта в цифровой среде», <https://rscf.ru/project/24-18-01125/>.

СПИСОК ЛИТЕРАТУРЫ

1. Агафонов А.А., Хусаинов А.Р., Устин П.Н, Попов Л.М. Большие языковые модели как инструмент прогнозирования типов поведения личности // Российский психологический журнал. 2026. № 23(2) (в печати).
2. Zhou Y., Muresanu A.I., Han Z., et al. Large Language Models Are Human-Level Prompt Engineers. arXiv preprint arXiv:2211.01910. 2023. <https://doi.org/10.48550/arXiv.2211.01910> (дата обращения: 15.05.2026).
3. Yang C., Wang X., Lu Y., et al. Large Language Models as Optimizers. arXiv preprint arXiv:2309.03409. 2024. <https://doi.org/10.48550/arXiv.2309.03409> (дата обращения: 15.05.2026).
4. Khattab O., Singhvi A., Maheshwari P., et al. DSPy: Compiling Declarative Language Model Calls into Self-Improving Pipelines. arXiv preprint arXiv:2310.03714. 2023. <https://doi.org/10.48550/arXiv.2310.03714> (дата обращения: 15.05.2026).

5. *Brown T.B., Mann B., Ryder N., et al.* Language Models are Few-Shot Learners // *Advances in Neural Information Processing Systems*. 2020. Vol. 33. P. 1877–1901.
6. *Reynolds L., McDonell K.* Prompt Programming for Large Language Models: Beyond the Few-Shot Paradigm // *Extended Abstracts of the 2021 CHI Conference on Human Factors in Computing Systems*. 2021. P. 1–7.
<https://doi.org/10.1145/3411763.3451760>
7. *Christiano P.F., Leike J., Brown T.B., et al.* Deep Reinforcement Learning from Human Preferences // *Advances in Neural Information Processing Systems*. 2017. Vol. 30. P. 4299–4307.
8. *Ouyang L., Wu J., Jiang X., et al.* Training Language Models to Follow Instructions with Human Feedback // *Advances in Neural Information Processing Systems*. 2022. Vol. 35. P. 27730–27744.
9. *Rafailov R., Sharma A., Mitchell E., et al.* Direct Preference Optimization: Your Language Model is Secretly a Reward Model // *Advances in Neural Information Processing Systems*. 2023. Vol. 36. P. 53728–53741.
10. *Wei J., Wang X., Schuurmans D., et al.* Chain-of-Thought Prompting Elicits Reasoning in Large Language Models // *Advances in Neural Information Processing Systems*. 2022. Vol. 35. P. 24824–24837.
11. *Davidson T., Warmusley D., Macy M., Weber I.* Automated Hate Speech Detection and the Problem of Offensive Language // *Proceedings of the 11th International AAAI Conference on Web and Social Media*. 2017. P. 512–515.
<https://doi.org/10.1609/icwsm.v11i1.14955>
12. *Zampieri M., Malmasi S., Nakov P., et al.* Predicting the Type and Target of Offensive Posts in Social Media // *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. 2019. P. 1415–1420. <https://doi.org/10.18653/v1/N19-1144>
13. *Kolhatkar V., Wu H., Cavasso L., et al.* The SFU Opinion and Comments Corpus: A Corpus for the Analysis of Online News Comments // *Corpus Pragmatics*. 2020. Vol. 4, No. 2. P. 155–190. <https://doi.org/10.1007/s41701-019-00065-w>
14. *Park G., Schwartz H.A., Eichstaedt J.C., et al.* Automatic Personality Assessment through Social Media Language // *Journal of Personality and Social Psychology*. 2015. Vol. 108, No. 6. P. 934–952. <https://doi.org/10.1037/pspp0000020>

15. Tibshirani R. Regression Shrinkage and Selection via the Lasso // Journal of the Royal Statistical Society. Series B (Methodological). 1996. Vol. 58, No. 1. P. 267–288. <https://doi.org/10.1111/j.2517-6161.1996.tb02080.x>
 16. Efron B. Bootstrap Methods: Another Look at the Jackknife // The Annals of Statistics. 1979. Vol. 7, No. 1. P. 1–26. <https://doi.org/10.1214/aos/1176344552>
 17. Doshi-Velez F., Kim B. Towards a Rigorous Science of Interpretable Machine Learning. arXiv preprint arXiv:1702.08608. 2017. <https://doi.org/10.48550/arXiv.1702.08608> (дата обращения: 15.05.2026).
 18. Lipton Z.C. The Mythos of Model Interpretability // Communications of the ACM. 2018. Vol. 61, No. 10. P. 36–43. <https://doi.org/10.1145/3233231>
 19. Van Hee C., Lefever E., Hoste V. We Usually Don't Like Going to the Dentist: Using Common Sense to Detect Irony on Twitter // Computational Linguistics. 2018. Vol. 44, No. 4. P. 793–832. https://doi.org/10.1162/coli_a_00337
-

ПРИЛОЖЕНИЕ А

Итоговая инструкция, сгенерированная PromptAlign. Полный текст инструкции (после второй итерации), использованной для финального тестирования (тег <no_think> – служебная команда отключения режима расширенных рассуждений модели):

<no_think>

Определи тип поведения автора текста. Есть три класса:

- Деструктивный – прямые оскорбления, угрозы, унижения, призывы к разрушению, паника, агрессия, а также выражение безысходности, эмоционального страдания или беспомощности без попыток решения.
- Информативный – аналитика, факты, статистика, цифры, описание событий или проектов; НЕТ личной позиции автора и призывов к действию. Даже при наличии эмоциональной окраски, сарказма или критики, если нет предложений решений.
- Конструктивный – предложения решений, эмпатия, поддержка, призывы к созданию, спокойный тон. Сюда же относится критика с аргументацией и предложением улучшений, анализ причин и следствий с выводами, личные наблюдения с призывом к диалогу.

Обобщенные правила:

1. Если текст содержит факты, цифры, статистику или аналитические выводы

без явных предложений по решению проблемы – это Информативный, даже при наличии эмоций или критики.

2. Если текст содержит критику, аргументацию и предложение решения (даже резкое) – это Конструктивный, а не Деструктивный.

3. Если текст содержит эмоциональное страдание, безысходность, жалобы на безвыходность без конкретных шагов – это Деструктивный, даже при наличии просьбы о совете.

4. Если текст содержит развернутое сравнение, анализ причин и следствий, выводы или позитивную оценку развития – это Конструктивный, даже при элементах пересказа.

5. Важно: Конструктивный класс часто путают с Информативным. Критика + предложение решения = Конструктивный. Просто факты = Информативный.

Ответь ОДНИМ СЛОВОМ (Информативный / Деструктивный / Конструктивный).

Текст: {text}

ПРИЛОЖЕНИЕ Б

Список извлекаемых признаков. В табл. Б.1 перечислены детерминированные признаки, вычисляемые без обращения к LLM; в табл. Б.2 – LLM-признаки, извлекаемые отдельным запросом к модели. Признак `speech_act` кодируется унитарно (one-hot), непрерывные значения масштабируются в отрезок [0, 1].

Табл. Б.1. Детерминированные признаки.

Номер	Признак	Описание
1	<code>rb_exclamation</code>	доля восклицательных знаков/предложение
2	<code>rb_question</code>	доля вопросительных знаков/предложение
3	<code>rb_caps_ratio</code>	доля заглавных букв
4	<code>rb_negative_words</code>	частота отрицательных частиц
5	<code>rb_first_person</code>	доля местоимений 1-го лица
6	<code>rb_we_form</code>	доля форм «мы», «нам», «наш»
7	<code>rb_imperative</code>	доля императивных конструкций
8	<code>rb_avg_sent_len</code>	средняя длина предложения (слов)
9	<code>rb_text_len_log</code>	логарифм длины текста
10	<code>rb_ellipsis</code>	количество многоточий

11	rb_quotes	число кавычек
12	rb_numbers	доля числовых выражений
13	rb_url	наличие URL (бинарный)

Табл. Б.2. LLM-признаки.

Номер	Признак	Тип	Описание
1	has_aggression	бинарный	прямая агрессия
2	has_indirect_aggression	бинарный	косвенная/пассивная агрессия
3	has_empathy	бинарный	эмпатия, поддержка
4	has_solution	бинарный	наличие предлагаемого решения
5	has_facts	бинарный	наличие фактов/статистики
6	has_author_position	бинарный	выраженная авторская позиция
7	has_call_to_action	бинарный	призыв к действию
8	speech_act	категориальный	тип речевого акта (6 категорий)
9	irony_probability	непрерывный [0,1]	вероятность иронии
10	emotional_intensity	непрерывный [0,1]	эмоциональная интенсивность

ПРИЛОЖЕНИЕ В

Примеры текстов для few-shot базового уровня. Ниже приведен показательный пример деструктивного класса (на него ссылается разд. «Обсуждение»); полный набор из 9 примеров (по 3 на класс), использованный при формировании few-shot-инструкции, доступен в репозитории проекта.

Пример (деструктивный класс)

Летом прошлого года Ученый совет Российского государственного гуманитарного университета утвердил создание Высшей политической школы имени Ивана Ильина. Разобраться в назначении данной конторы непросто. Для этого достаточно прочитать цель из положения о школе: «Разработка и реализация нового подхода... с опорой на российскую цивилизационную идентичность и традиционные российские духовно-нравственные ценности». Ну толковая же контора? Кто поспорит? По крайней мере, на бумаге все очень верно и грамотно построено. В общем, все красиво... Не то слово. Красивее не бывает.

[Приложение В. Примеры текстов для few-shot базового уровня] (https://github.com/AlexanderAgathonov/PromptAlign-Code/blob/main/supplementary_materials/appendix_b_few_shot_examples.md).

ПРИЛОЖЕНИЕ Г

Правила, сгенерированные метаанализом. Метаанализ ошибок на разных итерациях сформировал набор до 20 правил на естественном языке (формат JSON с полями reasons и rule); в обобщенном виде они вошли в итоговую инструкцию (Приложение А). Полный список правил доступен в репозитории.

[Приложение Г. Правила, сгенерированные метаанализом]

(https://github.com/AlexanderAgathonov/PromptAlign-Code/blob/main/supplementary_materials/appendix_c_generated_rules.md).

PROMPTALIGN: AUTOMATIC ITERATIVE ALIGNMENT OF A LARGE LANGUAGE MODEL TO EXPERT ANNOTATIONS WITHOUT FINE-TUNING

A. A. Agathonov¹ [0000-0001-6924-6701], A. R. Khusainov² [0009-0002-1916-2109],

N. A. Prokopyev³ [0000-0003-0066-7465]

¹⁻³Kazan (Volga region) Federal University, Kazan, Russia

¹a.a.agathonov@gmail.com, ²ahat2182@gmail.com, ³nikolai.prokopyev@gmail.com

Abstract

Large language models are increasingly used in social and psychological sciences; however, their predictions are systematically biased relative to expert annotations (alignment shift), and attempts to manually refine instructions or to supply examples lead to a “blanket-pulling” effect – degradation of minority classes.

We propose PromptAlign, an automatic prompt-optimization pipeline that requires neither access to model weights nor human involvement in iterative correction, because the taxonomy and the feature space are specified by an expert a priori. The pipeline extracts hybrid linguistic and semantic features using the large language model itself, builds interpretable L1-regularized logistic regressions to localize disagreements with the expert, performs a meta-analysis of errors, and compiles new prompts while controlling the class balance through a class-imbalance index (Blanket Pulling Index, BPI). An experiment on a corpus of 541 Russian-language texts with three psychological classes showed an increase in macro-F1 on the held-out test set from 0.655 to 0.749 and in the minimum per-class F1 from 0.449 to 0.656 ($p > 0.99$ by a paired bootstrap test). The method significantly outperforms both the zero-shot

baseline and few-shot prompting. A component-contribution analysis confirms the critical role of semantic interpretation of features and of periodic prompt rebuilding. The final prompt contains five rules that reflect latent expert criteria; unlike the “black boxes” produced by automatic optimizers, these rules are available for inspection and correction by a specialist.

Keywords: *large language model, prompt optimization, alignment, interpretability, L1 regularization, bootstrap test, text classification, psychological annotation, surrogate model, class imbalance index, Russian-language corpus.*

REFERENCES

1. Agathonov, A.A., Khusainov, A.R., Ustin, P.N., & Popov, L.M. Large language models as a tool for predicting types of personal behavior // Russian Psychological Journal. 2026. № 23(2) (in print).
2. Zhou Y., Muresanu A.I., Han Z., et al. Large Language Models Are Human-Level Prompt Engineers. arXiv preprint arXiv:2211.01910. 2023. <https://doi.org/10.48550/arXiv.2211.01910> (дата обращения: 15.05.2025).
3. Yang C., Wang X., Lu Y., et al. Large Language Models as Optimizers. arXiv preprint arXiv:2309.03409. 2024. <https://doi.org/10.48550/arXiv.2309.03409> (accessed May 15, 2026).
4. Khattab O., Singhvi A., Maheshwari P., et al. DSPy: Compiling Declarative Language Model Calls into Self-Improving Pipelines. arXiv preprint arXiv:2310.03714. 2023. <https://doi.org/10.48550/arXiv.2310.03714> (accessed May 15, 2026).
5. Brown T.B., Mann B., Ryder N., et al. Language Models are Few-Shot Learners // Advances in Neural Information Processing Systems. 2020. Vol. 33. P. 1877–1901.
6. Reynolds L., McDonell K. Prompt Programming for Large Language Models: Beyond the Few-Shot Paradigm // Extended Abstracts of the 2021 CHI Conference on Human Factors in Computing Systems. 2021. P. 1–7. <https://doi.org/10.1145/3411763.3451760>
7. Christiano P.F., Leike J., Brown T.B., et al. Deep Reinforcement Learning from Human Preferences // Advances in Neural Information Processing Systems. 2017. Vol. 30. P. 4299–4307.

8. *Ouyang L., Wu J., Jiang X., et al.* Training Language Models to Follow Instructions with Human Feedback // *Advances in Neural Information Processing Systems*. 2022. Vol. 35. P. 27730–27744.
9. *Rafailov R., Sharma A., Mitchell E., et al.* Direct Preference Optimization: Your Language Model is Secretly a Reward Model // *Advances in Neural Information Processing Systems*. 2023. Vol. 36. P. 53728–53741.
10. *Wei J., Wang X., Schuurmans D., et al.* Chain-of-Thought Prompting Elicits Reasoning in Large Language Models // *Advances in Neural Information Processing Systems*. 2022. Vol. 35. P. 24824–24837.
11. *Davidson T., Warmesley D., Macy M., Weber I.* Automated Hate Speech Detection and the Problem of Offensive Language // *Proceedings of the 11th International AAAI Conference on Web and Social Media*. 2017. P. 512–515.
<https://doi.org/10.1609/icwsm.v11i1.14955>
12. *Zampieri M., Malmasi S., Nakov P., et al.* Predicting the Type and Target of Offensive Posts in Social Media // *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. 2019. P. 1415–1420. <https://doi.org/10.18653/v1/N19-1144>
13. *Kolhatkar V., Wu H., Cavasso L., et al.* The SFU Opinion and Comments Corpus: A Corpus for the Analysis of Online News Comments // *Corpus Pragmatics*. 2020. Vol. 4, No. 2. P. 155–190. <https://doi.org/10.1007/s41701-019-00065-w>
14. *Park G., Schwartz H.A., Eichstaedt J.C., et al.* Automatic Personality Assessment through Social Media Language // *Journal of Personality and Social Psychology*. 2015. Vol. 108, No. 6. P. 934–952. <https://doi.org/10.1037/pspp0000020>
15. *Tibshirani R.* Regression Shrinkage and Selection via the Lasso // *Journal of the Royal Statistical Society. Series B (Methodological)*. 1996. Vol. 58, No. 1. P. 267–288. <https://doi.org/10.1111/j.2517-6161.1996.tb02080.x>
16. *Efron B.* Bootstrap Methods: Another Look at the Jackknife // *The Annals of Statistics*. 1979. Vol. 7, No. 1. P. 1–26.
<https://doi.org/10.1214/aos/1176344552>
17. *Doshi-Velez F., Kim B.* Towards a Rigorous Science of Interpretable Machine Learning. arXiv preprint arXiv:1702.08608. 2017.
<https://doi.org/10.48550/arXiv.1702.08608> (accessed May 15, 2026).

18. Lipton Z.C. The Mythos of Model Interpretability // Communications of the ACM. 2018. Vol. 61, No. 10. P. 36–43. <https://doi.org/10.1145/3233231>

19. Van Hee C., Lefever E., Hoste V. We Usually Don't Like Going to the Dentist: Using Common Sense to Detect Irony on Twitter // Computational Linguistics. 2018. Vol. 44, No. 4. P. 793–832. https://doi.org/10.1162/coli_a_00337

СВЕДЕНИЯ ОБ АВТОРАХ



АГАФОНОВ Александр Алексеевич – к. ф.-м. н., доцент, доцент кафедры высшей математики и математического моделирования Института математики и механики им. Н.И. Лобачевского, ФГАОУ ВО «Казанский (Приволжский) федеральный университет». Сфера научных интересов: обработка естественного языка, большие языковые модели, машинное обучение, искусственный интеллект в медицине, анализ поведения в цифровой среде, электронное образование.

Alexander Alekseevich AGATHONOV – Cand. Sc. (Physics and Mathematics), associate professor at the Department of Higher Mathematics and Mathematical Modeling, N.I. Lobachevsky Institute of Mathematics and Mechanics, Kazan Federal University. Scientific interests: natural language processing, large language models, machine learning, artificial intelligence in medicine, behavior analysis in digital environment, e-Learning.

email: a.a.agathonov@gmail.com

ORCID: 0000-0001-6924-6701



ХУСАИНОВ Ахат Рустемович – лаборант, ФГАОУ ВО «Казанский (Приволжский) федеральный университет». Сфера научных интересов: обработка естественного языка, машинное обучение, искусственный интеллект.

Akhat Rustemovich KHUSAINOV – laboratory assistant, Kazan Federal University. Scientific interests: natural language processing, machine learning, artificial intelligence.

email: khusainov.akhat@mail.ru

ORCID 0009-0002-1916-2109



ПРОКОПЬЕВ Николай Аркадиевич – к. т. н., доцент кафедры информационных систем Института вычислительной математики и информационных технологий, ФГАОУ ВО «Казанский (Приволжский) федеральный университет». Научный сотрудник Института прикладной семиотики Академии наук РТ. Сфера научных интересов: обработка естественного языка, искусственный интеллект, электронное образование.

Nikolai Arkadievich PROKOPYEV – Cand. Sc. (Technology), associate professor at the Department of Information Systems, Institute of Computational Mathematics and Information Technologies, Kazan Federal University. Researcher at the Institute of Applied Semiotics of the Academy of Sciences of the Republic of Tatarstan. Scientific interests: natural language processing, artificial intelligence, e-Learning.

email: nikolai.prokopyev@gmail.com

ORCID: 0000-0003-0066-7465

Материал поступил в редакцию 24 июня 2026 года