ЦИФРОВЫЕ ТЕХНОЛОГИИ | DIGITAL TECHNOLOGIES

Искусственный интеллект в Юго-Восточной Азии: как развиваются исследования больших языковых моделей в 2024–2025 годах[1]

Artificial Intelligence in Southeast Asia: The Development of Large Language Model Research in 2024–2025[2]

Гарин Артем Алексеевич

Кандидат исторических наук, руководитель Лаборатории цифровых исследований современного Востока Института востоковедения Российской академии наук

a.garin@ivran.ru

Artyom A. Garin

PhD (History), Head of the Digital Research Laboratory of Contemporary East at the Institute of Oriental Studies of Russian Academy of Sciences

ORCID: 0000-0003-4677-7221

В обзоре рассматривается, как страны Юго-Восточной Азии в 2024–2025 гг. развивали и применяли искусственный интеллект, прежде всего большие языковые модели (БЯМ). Основное внимание уделяется адаптации ИИ под местные языки, культуру и практические задачи. Показаны примеры из Вьетнама, Таиланда, Индонезии, Малайзии и Сингапура: создание моделей и датасетов для национальных языков, разработка систем для университетов и образовательных сервисов, а также попытки сделать ответы ИИ более точными, безопасными и социально приемлемыми. Отдельно обсуждаются вопросы экономии ресурсов. Статья показывает, что в Юго-Восточной Азии постепенно отходят от использования универсальных глобальных моделей и движутся к локальным, более прикладным решениям, ориентированным на узкие профессиональные задачи.

Ключевые слова: искусственный интеллект, большие языковые модели, Юго-Восточная Азия, RAG, Вьетнам, Таиланд, Индонезия, Малайзия, Сингапур

This review examines how countries in Southeast Asia developed and applied artificial intelligence in 2024–2025, with a particular focus on large language models. The main emphasis is on how AI systems are adapted to local languages, cultural contexts, and practical use cases. Examples from Vietnam, Thailand, Indonesia, Malaysia, and Singapore illustrate efforts to build language-specific models and datasets, develop systems for universities and educational services, and improve the accuracy, safety, and social appropriateness of AI-generated responses. The review also shows that the Southeast Asia is gradually moving away from universal global models toward localized, more application-oriented AI-solutions tailored to professional needs.

Keywords: Artificial Intelligence, Large Language Models, LLMs, Southeast Asia, RAG, Chatbots, Vietnam, Thailand, Indonesia, Malaysia, Singapore

[1] Работа выполнена в рамках государственного задания Министерства науки и высшего образования Российской Федерации (тема № FMNN-2024-0001 «Лаборатория цифровых исследований современного Востока»)

[2] The research was carried out within the state assignment of the Ministry of Science and Higher Education of the Russian Federation (theme No. FMNN-2024-0001 "Digital Research Laboratory of Contemporary East IOS RAS")

Развитие искусственного интеллекта в странах Юго-Восточной Азии

В последние годы Юго-Восточная Азия (ЮВА) превратилась в один из самых динамичных регионов мира по развитию искусственного интеллекта. Государства АСЕАН выстраивают ИИ-экосистемы, способные решать узкоспециализированные задачи. Ярким примером лидерства выступают Вьетнам, Таиланд, Индонезия, Малайзия и Сингапур. Все они двигаются к созданию локальных ИИ-решений.

Развитие технологий искусственного интеллекта, особенно в области больших языковых (LLM[3]) и рассуждающих (LRM[4]) моделей, требует постоянного отслеживания самых свежих научных достижений. Год в мире ИИ сегодня равен нескольким годам в других областях: модели, архитектуры и методы устаревают с невероятной скоростью. Именно поэтому для проведения актуального обзора в этой статье сделан осознанный выбор в пользу анализа препринтов научных работ, опубликованных на платформе arXiv [5] в 2024–2025 годах. arXiv служит агрегатором самых передовых и еще не формализованных в рецензируемых журналах идей. Исследования появляются здесь на месяцы, а иногда и на годы раньше их официальной публикации, что позволяет анализировать самые актуальные тенденции в ИИ-отрасли.

Вьетнам

Почти полный исследовательский цикл вокруг больших языковых моделей сложился во Вьетнаме: от сбора и очистки данных до архитектур для чат-ботов, которые должны отвечать строго по фактам. В стране формируют собственную ИИ-экосистему, которая умеет работать на вьетнамском языке, решать задачи университетов, студентов, программистов и исследователей.

Хорошее представление о развитии ИИ-систем во Вьетнаме дает препринт **«На пути к комплексным вьетнамским системам Retrieval-Augmented Generation и большим языковым моделям»** (Towards Comprehensive Vietnamese Retrieval-Augmented Generation and Large Language Models) [3]. Специалисты решили создать модель, работающую на больших, разнообразных и чистых материалах на вьетнамском языке. Для этого они собрали гигантские корпуса данных. Во-первых, это десятки миллионов материалов, очищенных и структурированных так, чтобы модель могла их «переварить». Исследователи сделали датасеты, где каждая новость записана в виде «заголовок — короткое резюме — полный текст». Подготовлен и большой набор для классификации новостей по темам, добавлен специальный набор инструкций, на которых можно обучать модели лучше следовать пользовательским запросам. Во-вторых, созданы и разговорные корпусные наборы для диалогов и ролевых сценариев. Важную роль в создании ИИ-модели на вьетнамском языке играет токенизатор — это инструмент, который разбивает текст на небольшие смысловые части (токены). Эти части могут быть словами, их фрагментами или символами и используются моделью для преобразования текста в числовое представление (векторы). Если модель токенизирует вьетнамские тексты так же, как английские, получается избыточно

[3] Large Language Model
[4] Large Reasoning Model
[5] https://arxiv.org/

много токенов, и обработка той же фразы обходится дороже. Авторы статьи обучили токенизаторы специально под вьетнамский. В результате количество токенов, необходимых для обработки текста, сократилось почти вдвое. Это экономит деньги и ускоряет работу, что особенно важно там, где ресурсы ограничены. На основе таких данных авторы дообучают вьетнамские версии моделей LLaMA2[6] и выкладывают их в публичный доступ, чтобы ими могли пользоваться университеты и компании.

В исследовании **«Оценка способности больших языковых моделей к генерации данных для проверки фактов на вьетнамском языке»** (Evaluating Large Language Model Capability in Vietnamese Fact-Checking Data Generation) [1] рассматривалось, могут ли большие языковые модели автоматически создавать обучающие данные для задачи фактчекинга на вьетнамском языке. *Фактчекинг* — это проверка достоверности утверждений на основе нескольких источников информации. Для низкоресурсных языков, таких как вьетнамский, не хватает размеченных данных, что затрудняет разработку автоматических систем проверки фактов. Авторы использовали статьи из вьетнамской Википедии (иностранный владелец ресурса «Википедия» нарушает закон РФ), из которых выбрали несколько связанных предложений. На их основе модели должны были сгенерировать утверждение и присвоить ему одну из трех меток: «Подтверждено» (утверждение соответствует доказательствам), «Опровергнуто» (утверждение противоречит доказательствам) или «Недостаточно информации» (на основе доказательств нельзя сделать вывод). Особенность задачи в том, что утверждение должно не просто копировать текст, а обобщать и синтезировать информацию из нескольких предложений, что требует развитых навыков понимания и логики. В эксперименте участвовали пять моделей: GPT-3.5, Gemini, Llama2, Qwen и специальная вьетнамская модель Vistral. Для улучшения их работы применялся метод LoRA (Low-Rank Adaptation) — техника эффективной настройки больших моделей. Вместо полного переобучения модели, которое требует огромных вычислительных ресурсов, LoRA добавляет к ее слоям небольшие обучаемые матрицы. Это позволяет адаптировать модель к конкретной задаче (например, генерации фактчекинг данных на вьетнамском) с минимальными затратами, сохраняя при этом все ранее полученные знания модели. Результаты показали, что модели способны автоматически создавать данные для фактчекинга, что снижает стоимость и время по сравнению с ручной разметкой. Например, использование модели Vistral сократило стоимость примерно в 20 раз по сравнению с созданием датасета человеком. Однако качество сгенерированных данных все еще уступает человеческому, особенно в аспектах логической связности, смысловой точности и способности к абстракции. Наиболее сложной для моделей оказалась метка «Недостаточно информации»: модели либо допускали «галлюцинации» (придумывали факты, не вытекающие из доказательств), либо, наоборот, были излишне осторожными.

Вместе с тем предпринимаются попытки научить модель не просто писать код, а исправлять свои собственные ошибки. В работе **«Эмпирическое исследование самокорректирующих больших языковых моделей для генерации кода в области науки о данных»** (An Empirical Study on Self-correcting Large Language Models for Data Science Code Generation) [2] представлен новый метод,

[6] Создана компанией META, признанной экстремистской на территории Российской Федерации.

названный CoT-SelfEvolve. Исследователи решили улучшить способность больших языковых моделей генерировать корректный и работоспособный код, особенно для задач анализа данных с использованием программных библиотек NumPy, Pandas и Scikit-learn. Когда большие языковые модели генерируют код, они допускают ошибки — синтаксические, логические или связанные с неверным использованием библиотек. Исправлением таких ошибок занимается человек-разработчик. Вьетнамские авторы предлагают научить модель самостоятельно исправлять свои ошибки на основе обратной связи от выполнения кода. Сначала модель получает описание задачи и, используя знания из обсуждений на веб-портале разработчиков StackOverflow[7], генерирует первый вариант кода. StackOverflow используется как источник практических решений и типичных ошибок, что делает начальный код более осмысленным. Затем сгенерированный код проверяется на ошибки, а после запускается в тестовой среде. Если код не проходит тесты, система фиксирует ошибки— например, сообщения об отсутствии переменной или неверном результате. Ошибки передаются обратно в модель, которая анализирует их с помощью цепочки рассуждений (Chain of Thought, CoT). CoT — это метод, при котором модель не просто выдает ответ, а шаг за шагом объясняет, как она пришла к решению, подобно тому, как рассуждает человек. Например, модель может последовательно: понять условие задачи, найти место ошибки в коде, определить причину сбоя и предложить конкретное исправление. На основе этого анализа модель переписывает код, и процесс повторяется до успешного выполнения или исчерпания попыток. Иначе говоря, Chain of Thought — подход, при котором модель разбивает сложную задачу на последовательные логические шаги. Например, вместо того чтобы сразу написать код для сортировки данных, модель может сначала описать, как нужно прочитать данные, затем выбрать метод сортировки, а потом реализовать его. Это делает процесс генерации кода более осмысленным и контролируемым. Эксперименты на наборе данных DS-1000[8], содержащем 1000 задач по дата-сайенс, показали, что CoT-SelfEvolve превосходит обычные БЯМ. Точность генерации кода увеличилась с 14% при одной попытке до 83% после пяти итераций исправлений. Наибольший скачок качества наблюдался между первой и второй попыткой, что подчеркивает важность обратной связи и самоисправления. Предполагается, что новый подход позволит создать «ассистента-разработчика», который пишет код, учится на своих ошибках, постепенно улучшая качество своих решений. Это особенно полезно в областях, где код должен быть точным и эффективным, например, в анализе данных и машинном обучении. Метод может быть интегрирован в среды непрерывной разработки (CI/CD[9]), автоматически исправляя ошибки в ходе тестирования.

В работе **«#URAG: реализация унифицированной гибридной системы RAG для точных ответов в чат-ботах по вопросам поступления в вузы — пример реализации в Технологическом ун-те Хошимина»** (URAG: Implementing a Unified Hybrid RAG for Precise Answers in University Admission Chatbots — A Case Study at HCMUT) [4] представлена гибридная система, которая помогает чат-боту

[7] https://stackoverflow.com

[8] https://github.com/xlang-ai/DS-1000

[9] Continuous Integration/Continuous Delivery – разработки программного кода методом непрерывной интеграции изменений разработчика в продукт и его поставки пользователю

для абитуриентов давать точные ответы, строго опираясь на официальные источники. Большие языковые модели могут выдавать неточные или вымышленные ответы, особенно на важные вопросы о поступлении — например, о сроках, баллах или условиях зачисления. Это критично, так как ошибка может повлиять на решение абитуриента. Авторы предлагают двухуровневую систему URAG (Unified RAG), которая работает как сотрудник приемной комиссии. Система сначала ищет точный ответ в заранее подготовленной базе часто задаваемых вопросов. Эта база содержит оригинальные вопросы и ответы, а также обогащена с помощью механизма URAG-F, который создает множество перефразированных вариантов одних и тех же вопросов. Это позволяет распознавать один и тот же запрос, сформулированный по-разному, и давать на него точный, неизменный ответ. Если в FAQ подходящего ответа нет, система переходит к поиску в базе официальных документов университета: правилах приема, описаниях программ, приказах. Документы предварительно обработаны механизмом URAG-D (они разбиты на смысловые блоки, каждый из которых снабжен кратким заголовком-резюме). Это позволяет найти именно тот фрагмент, который относится к вопросу, и сгенерировать ответ на его основе. Если и в документах информации нет, чат-бот выдает ответ, сгенерированный языковой моделью, но обязательно добавляет предупреждение, что информация может быть неточной. В эксперименте на реальных вопросах от школьников система URAG, использующая относительно небольшую локальную модель, показала более высокую точность (62,8%), чем мощные коммерческие модели вроде GPT-4o (54,4%), Claude 3.5 Sonnet (50,2%) и Gemini 1.5 Pro (44%). Ключевая причина — URAG не пытается придумывать ответ, а строго следует подготовленным данным. Особенно она преуспела в ответах на фактологические вопросы (даты, требования, цифры). Система интегрирована в чат-бот Технологического университета Хошимина (HCMUT) и использовалась абитуриентами. Пик обращений пришелся на периоды, предшествующие вступительной кампании.

Новые инструменты тестируют и для Вьетнамского национального университета в Ханое (Vietnam National University). В работе **«VietEduFrame: использование больших языковых моделей для управления образовательными процессами на вьетнамском языке в условиях ограниченных ресурсов»** (VietEduFrame: Using Large Language Models for Education Managements in Vietnamese with Low Resources) [6] авторы рассматривают применение больших языковых моделей в управлении образовательным процессом. Речь идет о приказах, правилах обучениях и административных документах. На проблему обратили внимание из-за дефицита вычислительных ресурсов, отсутствия готовых размеченных корпусов данных и высокой вариативности местных университетских документов. Вместо попытки обучать крупную модель с нуля исследователи предложили сочетать синтетическую генерацию данных, частичную ручную валидацию и дообучение существующих БЯМ под задачу доступа к регламентации образовательных процессов. В качестве исходного материала авторы задействовали официальные документы Вьетнамского национального университета в Ханое, которые структурировали в формате «контекст — вопрос — ответ», что позволило превратить разрозненные нормативные документы в единый машиночитаемый корпус. Технически VietEduFrame опирается на уже существующие языковые модели (Bloom и Vistral), которые дообучаются с использованием LoRA, что

радикально снижает требования к памяти и времени обучения. Эксперименты показали, что даже при ограниченных ресурсах возможно добиться приемлемого качества ответов: вьетнамская модель Vistral стабильно превосходит многоязычную Bloom по метрикам Exact Match и F1, а использование LoRA помогло сократить потребление GPU-памяти почти вдвое без критической потери точности. При этом авторы предупредили об ограничениях подхода: модель по-прежнему испытывает трудности с многошаговыми рассуждениями, плохо обрабатывает неоднозначные формулировки и чувствительна к качеству синтетических данных.

Авторы статьи **«Построение кросс-данных графа знаний для систем образовательной вопросно-ответной работы на основе больших языковых моделей»** (Cross-Data Knowledge Graph Construction for LLM-based Educational QA Systems**)** [5] предприняли попытку решить проблему галлюцинаций и недостатка актуальной/предметной информации у БЯМ в образовательной среде университета. Специалисты дополнили RAG-подход графом знаний (Knowledge Graph, KG). KG служит структурированным, обновляемым и проверяемым источником фактов, к которому модель обращается перед генерацией ответа. ИИ-решение для Технологического университета Хошимина состоит из трех частей. Сначала исследователи собрали все возможные источники информации — вопросы студентов из службы поддержки, документы с сайта, учебные расписания, правила университета. Вся эта информация раньше лежала вразнобой: в PDF, базах данных, на сайтах. Затем исследователи создали «умную карту знаний» университета, обозначив понятия: «курс», «преподаватель», «зачет», «стипендия», а связи между ними. Граф знаний обрабатывается, и машина может быстро в ней искать нужный ответ. Специалисты научили бота сначала смотреть в эту карту, а потом уже отвечать. Когда студент пишет вопрос, бот не сразу генерирует ответ, а сначала ищет в графе знаний факты: какие правила относятся к вопросу, какие документы, какие процедуры. Потом, имея точные данные на руках, формулирует четкий ответ. Самым сложном было научить сервис понимать, о чем спрашивают студенты. Вопросы всегда задаются по-разному: «хочу бросить предмет», «как отписаться от курса», «можно ли не ходить на пару». Ученые сгруппировали тысячи похожих вопросов, выделили общие темы и привязали их к соответствующим разделам правил. Система уже тестируется в университете.

На сегодняшний день во Вьетнаме создают основу для языковых моделей, оптимизированных под местный язык. Исследователи экспериментируют с автоматической генерацией данных, чтобы обойти дефицит размеченных корпусов. Вместе с тем изучаются способы заставить модель исправлять собственные ошибки, строятся прикладные системы, где LLM встроена в архитектуру, которая держит её в рамках официальных документов. Идет постепенное наращивание ИИ-инфраструктуры, включающей как корпуса текстов, так и реальные сервисы на базе искусственного интеллекта.

Таиланд

В Таиланде появился ряд новых работ, которые можно рассматривать как шаги к развитию тайских языковых моделей. Яркий пример — статья **«Представление мало представленных групп: оценки культурных и базовых возможностей для разработки крупных языковых моделей Таиланда»** (Representing

the Under-Represented: Cultural and Core Capability Benchmarks for Developing Thai Large Language Models**)** [7]. Авторы статьи поставили перед собой сложную задачу: понять, что значит хорошая языковая модель для тайского языка. Сегодня большинство больших языковых моделей обучаются и тестируются в англоязычном контексте. Даже когда модель формально поддерживает тайский язык, это обычно означает лишь, что она может генерировать грамматически правильные фразы. Но остается открытым вопрос умеет ли она рассуждать на тайском или понимать задачи, или говорить так, как принято в тайском обществе. Тайские датасеты традиционно сосредоточены на технических задачах вроде разметки текста или распознавания сущностей, но они слабо подходят для оценки современных LLM, которые используются как помощники. В результате невозможно понять, где модель действительно «умна», а где просто воспроизводит шаблоны без понимания культурного контекста.

Специалисты решили разделить задачу оценки на две разные части и создать два независимых тестовых набора данных: Thai-H6 и ThaiCLI. Thai-H6 нужен для ответа на вопрос, умеет ли модель думать и рассуждать на тайском языке. Под «мышлением» здесь понимаются логические выводы, работа с фактами, понимание школьной математики и неоднозначных формулировок. Это те же самые способности, которые обычно проверяют в англоязычных тестах вроде MMLU[10], ARC[11] или GSM8K[12]. Авторы не стали изобретать новые типы задач, а взяли шесть признанных международных систем оценки (бенчмарков) и полностью адаптировали их под тайский язык. Важно, что речь идет не о механическом переводе. В работу были вовлечены 43 носителя тайского языка, которые вручную проверяли каждое задание, исправляли ошибки и учитывали культурные особенности. В результате Thai-H6 состоит из шести крупных разделов, каждый из которых проверяет отдельную сторону «общего интеллекта» модели: логическое рассуждение, понимание контекста, фактические знания и умение решать задачи. Если модель показывает высокий результат на Thai-H6, это означает, что она способна решать интеллектуальные задачи на тайском языке, а не просто имитировать беглую речь.

Однако важно подчеркнуть, что языковая модель может хорошо справляться с логикой и математикой, но при этом говорить вещи, которые в тайском обществе звучат грубо, неэтично или даже опасно. Поэтому ThaiCLI устроен принципиально иначе, чем Thai-H6. Это тест на социальную и культурную адекватность. Он проверяет, как модель говорит о чувствительных для тайского общества темах: монархии, религии, политике, семейных традициях, повседневной жизни и экономике. Каждое задание в ThaiCLI представляет собой вопрос с двумя примерами ответа. Один из них считается культурно корректным (уважительный, аккуратный в формулировках, учитывает разные точки зрения и избегает резких обобщений). Второй — наоборот, показывает, как говорить не стоит (упрощение, субъективное мнение, неуважительные реплики или игнорирование социального контекста). Задача модели — сгенерировать собственный ответ, а затем он

[10] Massive Multitask Language Understanding – широкая проверка знаний и навыков больших языковых моделей на материале разных предметных областей и разных уровнях сложности.

[11] Abstraction and Reasoning Corpus – набор данных для оценки способностей ИИ к обобщению и рассуждению

[12] Grade School Math 8K —набор данных из примерно 8 тыс. текстовых математических задач для начальной школы для оценки способности больших языковых моделей

оценивается с точки зрения близости к «хорошему» культурному образцу. Иначе говоря, Thai-H6 и ThaiCLI вместе образуют более полную систему оценки ИИ-помощника. Первый отвечает за способность мыслить на тайском языке, второй — за умение быть частью тайского культурного пространства.

Чтобы измерить качество ответов в ThaiCLI, авторы использовали подход под названием «модель как судья». В этой роли выступала ИИ-модель GPT-4o. «Судье» показывали сам вопрос, затем ответ тестируемой модели, а рядом — два эталона: пример «хорошего» ответа (вежливого, нейтрального, культурно корректного) и пример «плохого» (упрощенного, неаккуратного или социально нечувствительного). После этого судья ставил оценку по шкале от 1 до 10 и кратко объяснял, почему ответ получился удачным или нет.

Что касается конкретных моделей, то Qwen2-72B или Llama-3.1-70B хорошо справлялись с Thai-H6. Они уверенно решали логические задачи, хорошо отвечали на вопросы из разных областей знаний и показывали высокий «интеллектуальный» уровень. Но когда дело доходило до ThaiCLI, картина менялась. Эти же ИИ-модели отвечали культурно корректно, но без особых нюансов. Их формулировки часто были нейтрально-общими, иногда слишком прямыми или явно заимствованными из англоязычных шаблонов. В результате их оценки по культурному тесту оказывались заметно ниже.

Парой к тому исследованию идет работа **«Способны ли универсальные большие языковые модели справиться с машинным переводом с английского на тайский?»** (Can General-Purpose Large Language Models Generalize to English-Thai Machine Translation?) [8]. Ее авторы решили выяснить, смогут ли универсальные большие языковые модели общего назначения эффективно заменить специализированные переводческие системы при работе с локальными языками в условиях ограниченных вычислительных ресурсов. Авторы сознательно уходили от абстрактного сравнения «кто умнее» и сфокусировались на реальных сценариях, с которыми сталкиваются медицинские учреждения, государственные структуры и локальные ИТ-команды в странах с другими языками, а не английским. Для исследования были выбраны две группы моделей. Первая — универсальная языковая модель Llama-3-8B, предназначенная для широкого круга задач: диалогов, анализа текста, генерации ответов и, в том числе, перевода. Вторая группа — модели семейства NLLB (No Language Left Behind), которые изначально разрабатывались как системы машинного перевода и обучались на парах языков, включая малоресурсные и локальные. Сравнивались два принципиально разных подхода — универсальная модель общего назначения и специализированный инструмент, оптимизированный под одну задачу.

Тайские специалисты построили эксперименты вокруг двух сценариев перевода. Первый сценарий представлял собой стандартную задачу: перевод английских предложений на тайский язык без дополнительных условий. Это самый распространенный и понятный случай, аналогичный использованию онлайн-переводчиков. Второй сценарий моделировал медицинский перевод со смешением языков внутри одного текста. В таких предложениях часть медицинских терминов, названий заболеваний и процедур оставалась на английском языке, в то время как остальной текст переводился на тайский. Этот сценарий принципиально сложнее, поскольку требовал понимания смысла и строгого соблюдения правил, так как некоторые слова переводить нельзя.

Оба набора данных выровняли по объему, чтобы исключить влияние количества примеров на итоговое качество. Это помогло сделать сравнение моделей корректным и сосредоточиться именно на различиях в их поведении и архитектуре. Отдельное внимание специалисты уделили условиям ограниченных вычислительных ресурсов. В реальных системах модели часто приходится сжимать, чтобы они могли работать на доступном оборудовании. Этот процесс называется квантованием, когда числовые параметры модели хранятся с меньшей точностью, что уменьшает объем памяти и ускоряет вычисления, но потенциально ухудшает качество. Авторы последовательно тестировали модели в разных режимах квантования — от относительно мягкого (8 бит) до крайне агрессивного (2 бита). Оказалось, что мультизадачная модель Llama-3-8B показала приемлемое качество перевода в 8-битном виде, однако при дальнейшем сжатии качество резко снижалось. На 4-битном уровне перевод начинал искажать смысл, а при 2-битном квантовании модель фактически переставала выполнять задачу перевода, выдавая бессвязный или ошибочный текст. Иначе говоря, модель чувствительна к жестким ограничениям по памяти и точности вычислений.

Специализированные модели семейства NLLB, несмотря на меньший размер, оказались устойчивее. Они сохраняли работоспособность и стабильное качество перевода при тех же ресурсных ограничениях и превосходили Llama-3-8B по ключевым метрикам качества. Это подтвердило гипотезу, что созданные исключительно для перевода модели лучше переносят сжатие и менее склонны к деградации при работе на слабом оборудовании.

Показательным стал сценарий с сохранением английских терминов в медицине. Универсальная модель Llama-3 в ряде случаев показывала неплохие результаты по метрикам, допускающим вариативность формулировок. Она часто правильно улавливала общий смысл и переформулировала текст своими словами. Однако именно там, где требовалась строгая точность (сохранение терминов и структуры), она чаще ошибалась. Универсальная модель склонна «пересказывать» текст, а не следовать формальным ограничениям перевода. Напротив, модели NLLB более жестко следовали структуре исходного текста и правилам задачи. Подобное поведение модели критически важно для медицинских и технических документов.

Исследователи также выяснили, что смешанный англо-тайский формат текста мог частично смягчать деградацию качества в сильно квантованных моделях. Предполагается, что английские слова и термины представлены в модели более устойчиво, чем записанные сложными тайскими символами, и поэтому сохранение части текста на английском снижает общий уровень искажений.

Новые исследования тайских ученых показывают, что универсальность моделей не всегда дает преимущество. Для ограниченных ресурсов и локальных задач специализированный инструмент может оказаться одновременно и точнее, и дешевле, и надежнее. Универсальные модели хороши как платформа, но, когда речь идет о реальных прикладных задачах, часто надо добавлять специализированные данные. В Таиланде, как и во Вьетнаме, идут попытки проверять интеллект ИИ-модели именно в контексте своей страны, как адаптировать существующие системы под локальные задачи. Еще одна задача — как избежать ошибок, которые могут нарушить культурные нормы.

Индонезия

Индонезийская исследовательская группа также решила выяснить, насколько современные языковые модели безопасны в местном культурном контексте. Работа получила название **«IndoSafety: безопасность в культурно-укорененных аспектах для больших языковых моделей на индонезийских языках»** (IndoSafety: Culturally Grounded Safety for LLMs in Indonesian Languages) [9]. Ее идея заключается в следующем: прежде чем требовать от моделей «этичного поведения», нужно понять, какие именно нормы и риски значимы для Индонезии. Большинство существующих тестов на «токсичность» и безопасность языковых моделей создавались в англоязычном контексте и отражают прежде всего ценности, страхи и табу западного мира. Эти тесты неплохо работают для английского языка, но при переносе в другие культурные среды начинают давать сбои. Они слабо чувствительны к региональным обычаям, не распознают локальный сленг, плохо улавливают религиозные и исторические подтексты и часто пропускают высказывания, которые для носителей другого языка звучат оскорбительно или социально опасно. Авторы анализируемой работы показали это на примере Индонезии. Даже локальные индонезийские модели, обученные на национальных данных, демонстрировали культурную уязвимость, когда ориентировались на англоязычные стандарты безопасности. Проблема здесь не в том, что встроенные фильтры безопасности просто не понимают, где именно пролегает граница допустимого в конкретной культурной среде. Для англоязычных систем такие темы выглядят нейтральными или экзотическими, тогда как для индонезийского общества связаны с идентичностью, верой и исторической памятью.

На этом фоне в Индонезии попытались вывести оценку безопасности языковых моделей из англо-центристского поля и привязать ее к реальным культурным условиям страны. Специалисты создали датасет IndoSafety, который предназначен для проверки и обучения языковых моделей с учетом индонезийского контекста. IndoSafety — вручную проверенный корпус запросов, отражающий реальные социальные и культурные моменты в Индонезии. Более того, Индонезия — многоязычная страна, люди используют разговорные формы, крупные региональные языки, такие как яванский, сунданский и минангкабау. Поэтому, если модель ведет себя безопасно только в формальном регистре речи, но начинает ошибаться в разговорном регистре или в речи на региональных диалектах, то в повседневном пользовании она потенциально небезопасна. Именно из-за этого датасет IndoSafety создали в пяти языковых вариантах.

Набор данных состоит из трех частей. Две из них предназначены для оценки — чтобы измерить, как часто и в каких ситуациях модель выдает потенциально опасные ответы. Третья часть используется для обучения моделей безопасному поведению и содержит примеры корректных, осторожных и социально приемлемых ответов на проблемные запросы. Среди чувствительных вопросов выделяют этнические и культурные стереотипы, трактовки исторических событий, вопросы, связанные с конкретными индонезийскими политическими и общественными фигурами. Отдельные опасения есть по темам сепаратизма, религиозной чувствительности и местных верований. В результате проделанной работы авторы выделили 19 типов риска, объединенных в шесть крупных категорий (дискриминация, «токсичность» и язык ненависти; вред, возникающий во взаимодействии человека и чат-бота; информационные угрозы; вредоносные и неэтичные

действия; вред от дезинформации; регионально-специфическая чувствительность).

Датасет создавали в два больших этапа. Сначала исследователи сосредоточились на классическом индонезийском языке и собрали набор IndoSafety-Eval-1, состоящий из 2514 запросов. Это нужно для проверки, как модель отвечает на потенциально опасные или чувствительные вопросы. Источники запросов были разными: одна часть представляла собой адаптацию уже существующих англоязычных тестов безопасности. Примеры на английском перерабатывались вручную: менялись реалии, имена, социальные роли, культурные отсылки и формулировки, чтобы запрос звучал естественно для индонезийского пользователя и действительно затрагивал релевантные местные темы. Вторая, не менее важная часть запросов была написана специально под индонезийский контекст. Для каждого типа риска авторы вручную формулировали вопросы, которые отражают именно те точки напряжения, где языковые модели чаще всего «спотыкаются»: религиозные практики, исторические сюжеты, этнические стереотипы, государственная идеология, сепаратизм и местные верования. Эти запросы не имели прямых аналогов в англоязычных наборах и стали оригинальным исследовательским материалом. Чтобы расширить вариативность формулировок и избежать однотипности, авторы дополнительно использовали модель QwQ-32B для генерации запросов по методу *few-shot*, то есть на основе небольшого числа вручную заданных примеров. Важный момент здесь заключается в выборе инструмента. Более строгие модели вроде GPT-4o или DeepSeek R1 для этой задачи оказались непригодны, поскольку их встроенные механизмы безопасности отказались генерировать чувствительный контент даже в исследовательских целях. QwQ-32B позволила получить черновые варианты запросов, которые затем проверялись вручную. Индонезийские специалисты подчеркивают, что каждый автоматически сгенерированный пример редактировался людьми, чтобы убрать языковые ошибки, неточности и культурные искажения. В результате было сформировано примерно 1000 «культурно чувствительных» запросов.

На втором этапе исследование вышло за рамки формального языка. Авторы создали многоязычную часть датасета — IndoSafety-Eval-2. Из большого корпуса первого этапа они отобрали 500 запросов так, чтобы были представлены все типы риска и все крупные категории вреда. Эти 500 запросов стали основой для многоязычного расширения. Каждый из них перевели на четыре дополнительные языковые формы: яванский, сунданский, минангкабау и разговорный индонезийский. Процесс перевода был двухступенчатым. Сначала использовался машинный перевод, а затем носители соответствующих языков вручную исправляли тексты. В итоге получилось 2500 параллельных примеров (одни и те же запросы, сформулированные в пяти языковых вариантах). Такая структура позволяет проверять, что отвечает модель и как меняется ее поведение при смене языка или регистра.

Оставшиеся 2014 запроса из первого этапа использовались для создания обучающей выборки IndoSafety-Train. Для каждого потенциально вредного запроса исследователи с помощью GPT-4o сгенерировали безопасный ответ. Под безопасным понимается не просто отказ, а структурированный, вежливый ответ,

который объясняет, почему запрос может быть проблемным, указывает на возможные риски и предлагает нейтральную или социально приемлемую альтернативу.

После создания датасета авторы перешли к эмпирической проверке реальных языковых моделей. Для эксперимента было выбрано десять LLM: закрытые коммерческие модели, такие как GPT-4o и Claude 3; открытые многоязычные модели семейств Qwen, Gemma и Llama; региональные модели Юго-Восточной Азии (Sailor2, SeaLLMs, SEA-LION); а также индонезийская модель SahabatAI. Оценка их оригинальных ответов проводилась с помощью GPT-4o в роли «цензора». Для каждой категории риска был подготовлен набор проверочных вопросов, например: содержит ли ответ оскорбление религии, поддерживает ли дискриминацию, подает ли спорную информацию как безусловный факт. Если хотя бы на один из этих вопросов ответ был положительным, ответ ИИ-модели признавался небезопасным. Чтобы убедиться, что автоматическая оценка не уходит слишком далеко от человеческого восприятия, часть результатов дополнительно сравнивалась с ручной разметкой.

Полученная картина оказалась далека от идеальной. Все протестированные модели время от времени выдавали вредные или некорректные ответы, но масштаб проблемы сильно различался. Среди закрытых систем Claude 3 показала более осторожное поведение, чем GPT-4o, хотя последняя лучше справлялась с региональными языками. Среди открытых моделей проблемной оказалась Sailor2: в некоторых тестах небезопасными признавались около трети ее ответов. На другом уровне находилась Gemma2, у которой доля вредных ответов в формальном и разговорном индонезийском языке не превышала 10%. Индонезийская SahabatAI особенно часто ошибалась именно в регионально чувствительных темах. Общими слабыми местами для всех моделей стали советы в кризисных ситуациях, дезинформация и вопросы, связанные с историческими и политическими чувствительностями.

На финальном этапе авторы взяли самую небезопасную модель — Sailor2-8B — и дообучили ее на выборке IndoSafety-Train, используя только формальных индонезийский язык. После этого модель снова прошла все тесты. Результаты показали резкое снижение числа вредных ответов во всех категориях и на всех языках. В блоке регионально чувствительных вопросов, например, количество небезопасных ответов в разговорном индонезийском сократилось примерно в десять раз.

Однако наблюдались и ограничения. Тесты состояли из прямых запросов — без хитрых многоходовых атак и без профессиональных техник промтинга для обхода систем безопасности ИИ-модели[13]. Однако все это не отменяет заслуг индонезийских специалистов, ведь до появления IndoSafety не было инструмента, который позволял бы оценивать безопасность модели в условиях индонезийской культуры и политики.

[13] Техники обхода (jailbreak) системы безопасности ИИ-моделей — это специальные методы составления промтов (запросов), призванные обойти встроенные в языковую модель ограничения безопасности и этики.

Малайзия

В препринте **«Диагностика неисправностей в электрических сетях с использованием больших языковых моделей»** (Fault Diagnosis in Power Grids with Large Language Model) [10] рассматривается интеллектуальная диагностика аварий в энергосистемах с использованием больших языковых моделей. Авторы пытались исходить из инженерного контекста. Дело в том, что современные энергосети — это сложные системы, на которые влияют много взаимосвязанных факторов. Классические системы диагностики в энергетике, как правило, работают по пороговому принципу: если параметр превышает допустимое значение, система фиксирует тревогу. Этот подход считается надежным, однако исследователи решили сфокусироваться на ситуациях, когда авария формируется постепенно.

Целью работы стала проверка гипотезы: могут ли большие языковые модели выступать в роли интеллектуального аналитического помощника для предотвращения аварий. Ключевой технический момент статьи — разработка специальной системы подсказок в промптах, которая заставляет модель рассуждать так, как это делает инженер-диагностик. Вместо абстрактного вопроса модели подавался структурированный текст, близкий по форме к инженерному отчету. В нем последовательно приводились измерения с оборудования (например, значения токов, напряжений и температур), затем выдержки из архива прошлых аварий и, наконец, описание конфигурации энергосети и характеристик оборудования. ИИ-модели задавали роль специалиста по диагностике энергосистем, после чего формулировали следующую задачу: определить возможную неисправность, подробно объяснить ход рассуждений и предложить практические рекомендации.

Для проверки подхода авторы собрали собственный датасет, ориентированный именно на задачу диагностики. Он включает три типа данных. Первый — числовые данные с узлов энергосистемы, отражающие текущее физическое состояние оборудования. Второй — исторические записи об авариях, содержащие информацию о причинах, условиях возникновения и последствиях неисправностей. Третий — текстовые описания структуры сети и отдельных компонентов. Все эти данные объединяются в единый текстовый контекст, который подается языковой модели без дополнительного обучения, только за счет правильно сформулированной подсказки.

Экспериментальная часть работы построена на сравнении нескольких способов взаимодействия с языковой моделью. Рассматривается стандартный запрос без специальных инструкций, уже упоминавшийся подход Chain-of-Thought, где модель просят рассуждать по шагам, и Tree-of-Thought (размышление по древесной схеме), в котором она анализирует несколько возможных вариантов причин аварии. Эти методы сравниваются с предложенным авторами инженерно-ориентированным видом подсказки. Оценка проводилась по нескольким критериям: диагностическая точность, качество объяснений, логика ответов и степень использования предоставленного контекста. Для оценки объяснимости использовалась модель GPT-4 в роли внешнего эксперта, который анализирует ответы модели как инженер-рецензент.

Результаты показали, что предложенный авторами метод стабильно превосходит стандартные способы взаимодействия с ИИ-моделью. Языковая модель

чаще определяла характер неисправности, реже противоречила сама себе и выдавала более понятные и аргументированные объяснения. Это сохранялось и в сложных сценариях, когда в системе присутствовали несколько неисправностей одновременно.

В заключении статьи подчеркивается, что проделанная работа не направлена на замену инженерных расчетов или автоматических защитных систем. Авторы предлагают рассматривать большие языковые модели как вспомогательного ИИ-ассистента, который помогает оператору энергосистемы быстрее понять ситуацию, связать текущие данные с прошлым опытом и принять обоснованное решение.

Сингапур

Исследователи в Сингапуре также пытаются сделать диалоговые системы и рассуждающие модели одновременно умнее, экономнее и безопаснее. Появляются работы, каждая из которых решает разные элементы этой задачи — долгосрочная память в диалоге, социальная динамика в многоагентных системах, проверка фактов в стиле человеческого поиска и оптимизация «размышляющих» моделей, чтобы они не утонули в собственных «мыслях».

В статье **«И снова здравствуй! Персонализированный агент для долгосрочного диалога на основе большой языковой модели»** (Hello Again! LLM-powered Personalized Agent for Long-term Dialogue) [11] сингапурские специалисты рассказали, как обеспечить долгосрочный диалог между человеком и языковой моделью без модификации самой модели. Авторы исходили из того, что современные LLM по своей природе хорошо работают внутри одного диалога, но не имеют механизма устойчивой памяти между сессиями и не поддерживают консистентность пользовательских и агентных характеристик во времени. Исследователи предложили решение под названием LD-Agent, которое представляет собой внешнюю модульную надстройку из трех компонентов. Первый компонент — модуль событийной памяти — отвечает за фиксацию и извлечение информации о прошлых взаимодействиях. Память принципиально разделена на краткосрочную и долговременную. Краткосрочная память содержит полный текст текущей сессии и используется как обычный контекст генерации. Долговременная память хранит не сами диалоги, а их сжатые представления в виде так называемых «саммари» (они автоматически генерируются моделью, преобразуются в текстовое описание события и одновременно кодируются в векторное представление). Поиск по долговременной памяти осуществляется гибридным образом: учитывается семантическая близость эмбеддингов, тематическое совпадение и временной фактор, придающий больший вес недавним событиям. Второй компонент — персонализированный модуль (модуль акторов) — предназначен для извлечения устойчивых характеристик пользователя и агента. Каждая новая реплика анализируется на предмет того, содержит ли она информацию, релевантную для долгосрочной персонализации пользователя или агента. Третий компонент агрегирует все источники информации: текущий контекст, извлеченные события из долговременной памяти и актуальные персональные данные, после чего формирует расширенный промпт для базовой модели. Эксперименты показали, что такая архитектура повышает связность диалога и качество персонализированных ответов независимо от выбранной базовой LLM. Основные ограничения метода

связаны с накоплением «шумной» информации в долговременной памяти и отсутствием механизма забывания.

Препринт **«Мультиагенты — это социальные группы (Multi-Agents)»** (Multi-Agents are Social Groups) [12] исследует поведенческий эффект взаимодействия человека с несколькими ИИ-агентами. Специалисты решили выяснить, влияет ли количество агентов на изменение мнения пользователя и субъективное ощущение социального давления при фиксированном содержании аргументов и стиле коммуникации. Для этого проведен контролируемый пользовательский эксперимент с 93 участниками. Каждый участник взаимодействовал либо с одним, либо с тремя, либо с пятью агентами. Во всех условиях агенты транслировали идентичные аргументы, различие заключается исключительно в числе источников. Затем измерялись позиции до и после диалога, субъективное ощущение давления, доверие к аргументам. Оказалось, что мнение менялось в работе с тремя ИИ-агентами. При пяти агентах усиливался эффект давления, приводя к сопротивлению пользователя. Основный технический вывод состоит в том, что участники не считают аргументы более правильными или качественными. Работает исключительно ощущение давления со стороны группы. Авторы подчеркнули, что эффект больше обусловлен социальной динамикой, а не когнитивной оценкой информации.

В работе **«EMULATE: Многоагентная инфраструктура для определения достоверности атомарных утверждений»** (EMULATE: A Multi-Agent Framework for Determining the Veracity of Atomic Claims) [13] предложили реальный человеческий процесс поиска и оценки информации от ИИ-моделей. Исследователи предложили многоагентную архитектуру, эмулирующую последовательные когнитивные шаги человека. EMULATE реализован как управляемый итеративный цикл, в котором каждый агент отвечает за одну узкую функцию. Отдельные агенты генерируют поисковые запросы, ранжируют результаты по релевантности и надежности, читают документы, оценивают новизну информации и достаточность доказательной базы. Особенностью метода стал механизм отложенных документов: страницы, которые на раннем этапе невозможно интерпретировать из-за недостатка контекста, сохраняются и повторно анализируются на более поздних шагах. Процесс продолжается до тех пор, пока агент достаточности не определит, что доказательств достаточно для вынесения вердикта. Эксперименты проводились на трех бенчмарках для фактчекинга. EMULATE показал устойчивое превосходство над существующими системами, особенно в классе ложных утверждений, который традиционно является самым сложным. Авторы также рассказали, что архитектура сохраняет эффективность даже при использовании относительно слабой базовой языковой модели. Ограничения метода связаны с вычислительной стоимостью, зависимостью от качества веб-источников и фокусом исключительно на атомарных утверждениях.

В работе **«Эффективный логический вывод для больших моделей рассуждений»** (Efficient Inference for Large Reasoning Models) [14] систематизированы методы снижения вычислительной стоимости вывода для больших рассуждающих моделей, использующих Chain-of-Thought. Авторы исходили из того, что рассуждения в виде длинных текстовых цепочек резко увеличивают количество токенов, время и потребление памяти, что делает такие модели плохо масштаби-

руемыми. На этом фоне исследователи ввели базовую таксономию из двух классов подходов. Первый класс — компактные Chain-of-Thought — сохраняет текстовую форму рассуждений, но сокращает их длину. Это достигается либо прямым сжатием цепочек, либо обучением на компактных примерах рассуждений, либо введением функций вознаграждения и наказания (штрафов), стимулирующих краткость. Второй класс — переносит рассуждение во внутреннее пространство скрытых состояний модели, так что наружу выводится только финальный ответ или краткий план. Последний метод позволил радикально снизить число токенов при сопоставимом качестве на математических и программных задачах. Однако авторы напомнили, что по мере сокращения или скрытия рассуждений становится все сложнее анализировать логику модели и выявлять потенциально небезопасные шаги. Дополнительным ограничением стало и то, что почти все существующие методы тестируются исключительно на формализованных задачах, тогда как для социальных и гуманитарных сценариев отсутствуют надежные метрики оценки.

Выводы

Юго-Восточная Азия меняет свое положение в мировой сфере искусственного интеллекта. Регион все чаще выступает как самостоятельный центр экспериментов с ИИ, ориентированных на местные языки, культуры и практические задачи. Дело в том, что универсальные модели плохо работают в локальных условиях и требуют глубокой настройки.

В целом ситуация с ИИ в Юго-Восточной Азии характеризуется переходом от заимствования к локализации. Страны региона все больше рассматривают ИИ как инструмент, который нужно адаптировать под язык, культуру, ресурсы и социальные риски. Этот подход делает развитие более устойчивым и применимым в реальной жизни.

Библиография | References

1. To L. T., Le H. T., Nguyen D. V.-T., Nguyen M. T., Nguyen T. T., Huynh T. V., Nguyen K. V. Evaluating Large Language Model Capability in Vietnamese Fact-Checking Data Generation [Электронный ресурс]. — https://arxiv.org/pdf/2411.05641, 2024.
2. Thai T. Q., Duc H. M., Thanh T. Q., Nguyen-Duc A. An Empirical Study on Self-correcting Large Language Models for Data Science Code Generation [Электронный ресурс]. — https://arxiv.org/abs/2408.15658, 2024.
3. Nguyen Q. D. et al. Towards Comprehensive Vietnamese Retrieval-Augmented Generation and Large Language Models [Электронный ресурс]. — https://arxiv.org/abs/2403.01616, 2024.
4. Nguyen L., Quan T. URAG: Implementing a Unified Hybrid RAG for Precise Answers in University Admission Chatbots – A Case Study at HCMUT [Электронный ресурс]. — https://arxiv.org/abs/2501.16276, 2025.
5. Bui T., Tran O., Nguyen P., Ho B., Nguyen L., Bui T., Quan T. Cross-Data Knowledge Graph Construction for LLM-based Educational QA Systems [Электронный ресурс]. — https://arxiv.org/abs/2404.09296, 2024.

6. Do M. D., Nguyen V. V., Dam C. T. VietEduFrame: Using Large Language Models for education managements in Vietnamese with low resources [Электронный ресурс]. — https://arxiv.org/abs/2501.15022v1, 2025.
7. Kim D., Lee S., Kim Y., Rutherford A., Park C. Representing the Under-Represented: Cultural and Core Capability Benchmarks for Developing Thai Large Language Models [Электронный ресурс]. — https://arxiv.org/abs/2410.04795, 2024.
8. Chiaranaipanich J. et al. Can General-Purpose Large Language Models Generalize to English-Thai Machine Translation? [Электронный ресурс]. — https://arxiv.org/abs/2410.17145, 2024.
9. Azmi M. F., Al Kautsar M. D., Wicaksono A. F., Koto F. IndoSafety: Culturally Grounded Safety for LLMs in Indonesian Languages [Электронный ресурс]. — https://arxiv.org/abs/2506.02573, 2025.
10. Liu J., Rahman A. Fault Diagnosis in Power Grids with Large Language Model [Электронный ресурс]. — https://arxiv.org/abs/2407.08836, 2024.
11. Li H., Yang C., Zhang A., Deng Y., Wang X., Chua T.-S. Hello Again! LLM-powered Personalized Agent for Long-term Dialogue [Электронный ресурс]. — https://arxiv.org/abs/2406.05925, 2024.
12. Song T., Tan Y., Zhu Z., Feng Y., Lee Y.-C. Multi-Agents are Social Groups: Investigating Social Influence of Multiple Agents in Human-Agent Interactions [Электронный ресурс]. — https://arxiv.org/abs/2411.04578, 2024.
13. Hong S., Luo M., Wan X. EMULATE: A Multi-Agent Framework for Determining the Veracity of Atomic Claims by Emulating Human Actions [Электронный ресурс]. — https://arxiv.org/abs/2505.16576v2, 2025.
14. Liu Y., Wu J., He Y., Gao H., Chen H., Bi B., Gong R., Zhang J., Huang Z., Hooi B. Efficient Inference for Large Reasoning Models: A Survey [Электронный ресурс]. — https://arxiv.org/abs/2503.23077, 2025.