МОДУЛЬ ОБЪЯСНИМОСТИ В МУЛЬТИАГЕНТНОЙ СИСТЕМЕ ДЕТЕКЦИИ ИИ-ТЕКСТОВ

Мельников Д.А., старший преподаватель,
ФГБОУ ВО «МИРЭА – Российский технологический университет», г. Москва, Россия

Аннотация. В данной статье представлено исследование модуля объяснимости, встроенного в мультиагентную систему детекции текстов, генерируемых искусственным интеллектом, в образовательной области. Описаны функциональные блоки модуля, включая Селектор (пороговая фильтрация аргументов), Адаптер (терминологическая и стилистическая адаптация под профиль преподавателя) и Генератор (синтез текста для базового, детального и экспертного уровней). Для детализации работы модуля приведена диаграмма потоков данных.

Ключевые слова: мультиагентные системы, агент, детекция ИИ-текстов, искусственный интеллект.

С появлением больших языковых моделей задача автоматической детекции текстов, сгенерированных искусственным интеллектом (ИИ), стала критически важной в сфере образования [1, 2, 3]. Несмотря на то, что большинство таких продуктов показывают высокую точность, их решения не всегда могут быть доступны преподавателям. Эта проблема часто возникает в ситуациях, когда преподаватель не может объяснить студенту, на каком основании его работа не прошла проверку на оригинальность, поскольку множество современных систем детекции выдают бинарный вердикт (есть в тексте ИИ или он отсутствует) и процент уверенности, но не объясняют, почему такое решение было принято.

Важность интегрирования модуля объяснимости в мультиагентную систему позволит преподавателю верифицировать решение, запросив

детальный отчет с исходными признаками, методами измерения и весами доверия к аргументам.

Выбранная система детекции ИИ-текстов состоит из четырех агентов. Агент-Аналитик выполняет функцию объективного измерения, извлекая из текста числовые признаки (перплексию, синтаксические, лексические и другие признаки[4]) с указанием доверительных интервалов. Агент-Обвинитель формулирует аргументы в пользу гипотезы «искусственный интеллект» и выбирает признаки с высоким обвинительным показателем. Агент-Защитник предлагает альтернативные интерпретации, учитывая контекст (дисциплину, уровень студента, историю работ) и генерирует контраргументы. Агент-Судья наблюдает полную историю диалога Обвинителя и Защитника, взвешивает силу аргументов (с использованием обученных весов доверия) и выносит финальный вердикт с определенной уверенностью. Стоит отметить, что именно Судья инициирует генерацию объяснения.

В основе проектируемого модуля объяснимости понимается способность системы не только сообщать результат классификации, но и воспроизводить полную логику принятия решения, включая:

- какие данные и признаки были использованы;
- какие альтернативные гипотезы рассматривались;
- почему одна из гипотез была признана более убедительной;
- какие неопределенности остались неразрешенными.

Модуль объяснимости разрабатывается как часть архитектуры мультиагентной системы детекции ИИ-текстов, опираясь на следующие функциональные принципы:

1. Решение системы должно быть прослеживаемо до исходных данных, действий агентов и принятых решений. Преподаватель должен иметь возможность получить полный отчет, начиная санализа конкретной части текста, заканчивая оценкойАналитика, Защитника и Обвинителя и вклада этой оценки в итоговое решение Судьи.

2. Модуль должен поддерживать разные уровни детализации объяснения в зависимости от контекста и запроса преподавателя. Можно выделить основные три уровня: базовый, детальный и экспертный. Базовый уровень может включать краткое резюме решения системы.Детальный уровень представляет из себя разбор по отельным агентам с предоставлением количественных показателей и аргументов каждого из них. Экспертный уровень может отражать полную цепочку принятия решения с доступом к исходным данным и внутренним представлениям компонентов.

3. В случае неточностей должен быть обеспечен интерактивный режим работы, позволяющий преподавателю задавать уточняющие вопросы и получать генерируемые разъяснения.

Для системы характерен, по большей мере, принцип внутренней объяснимости, где способность давать объяснения заложена в структуру агентов и процессы принятия решений. Такая объяснимость достигается благодаря нескольким ключевым архитектурным решениям. Во-первых, система включает четырех агентов с явными ролями, поведение которых определяется изначально интерпретируемыми правилами и таблицами. Во-вторых, в процессе диалога фиксируется полная история аргументов с указанием типа, весов доверия и ссылок на признаки. В-третьих, все механизмы внутри агентов (расчет признаков, формирование аргументов, вынесение взвешенного вердикта) являются открытыми для проверки.

На вход модуля поступают два вида информации: полная история диалогаH(History) между Агентом-Обвинителем и Агентом-Защитником (совокупность аргументов с их силой влияния и ссылками на признаки), итоговый вердиктV (Verdict) и степень уверенности C (Confidence), вычисленные Агентом-Судьей на основе взвешивания этих аргументов.

Формирование адаптированного под преподавателя объяснения требует решения трех самостоятельных, но взаимосвязанных задач, включая сокращение объема информации (выделение наиболее значимых аргументов), адаптацию текста к профилю пользователя и синтез связного итогового

объяснения, объединяющего вердикт, уверенность и ключевые аргументы. В соответствии с этим модуль объяснимости структурно разделен на три функциональных блока – Селектор, Адаптер и Генератор.

Главная задача Селектора состоит в сокращении объема данных путем отбора только тех аргументов, которые существенно повлияли на решение Судьи или могут быть полезны преподавателю при заданном уровне детализации. Селектор реализует принцип релевантности (чем выше требуемая детализация, тем больше аргументов сохраняется).

Адаптер преобразует технический язык аргументов в терминологию, соответствующую уровню подготовки преподавателя, а Генератор объединяет адаптированные аргументы, вердикт и уверенность в связное сообщение на естественном языке.

В таблице 1 кратко поясняются основные входные данные у каждого блока.

Таблица 1 – Входные данные блоков модуля объяснимости

Блок	Входные данные
Селектор	Полная история диалога *H* (тип аргумента, числовая сила влияния аргумента, список ссылок на признаки из Агента-Аналитика, номер итогового раунда и временная метка); финальный вердикт *V*(значение «ИИ» или «человек»); уверенность *C* (в значениях от 0 до 1); уровень детализации будущего объяснения*L*(1, 2 или 3 уровень); профиль преподавателя (пока не используется на этом этапе, но передается дальше)
Адаптер	Отфильтрованный список аргументов *A*; вердикт *V;* уверенность *C*; профиль преподавателя (уровень подготовки BEGINNER, INTERMEDIATE, EXPERT, предпочитаемый язык, изначальный список терминов, дисциплина для учета профессиональной специфики); уровень детализации *L*
Генератор	Список адаптированных аргументов *AdaptA*

	вердикт V; уверенность C; уровень детализации L

Логика работы Селектора состоит из следующих этапов:

1. Определяется τ(L) – порог силы аргумента, зависящий от уровня детализации L. Он устанавливается следующим образом:

– τ(0) – базовый уровень только с сильными аргументами,

– τ(1) – детальный уровень с аргументами средней силы,

– τ(2) – экспертный уровень со всеми аргументами без исключения.

Из H выбираются аргументы, для которых сила аргумента больше или равна τ(L). Если после фильтрациине остается ни одного аргумента, то Селектор автоматически выбирает два аргумента с наибольшей силой, даже если их сила ниже порога.

2. Сохраняется полная история аргументов для возможного аудита (просмотра всех аргументов, даже отфильтрованных) и повторного формирования объяснения без нового прогона агентов (например, при смене уровня детализации).

Адаптер работает по следующему принципу:

1. Таблица соответствий терминов загружается из конфигурационного файла, где для каждого исходного термина содержатся три варианта записи. Для начального уровня – это замена на более простые слова, для среднего – термин с кратким пояснением, для экспертного – оригинальный термин.

2. Текст каждого аргумента проходит через токенизацию. Для каждого токена проверяется, входит ли он в множество ключей таблицы. При совпадении он заменяется на соответствующий вариант в зависимости от уровня подготовки преподавателя. Также дополнительно могут адаптироваться термины на основе дисциплины и добавляться ссылки на научные исследования.

Наконец, Генератор выполняет следующие действия:

1. Выполняется предварительная подготовка данных, где аргументы разделяются на два списка – аргументы Защитника и Обвинителя.

2. При базовом (L=0) и детальном (L=1) уровнях генератор использует заранее подготовленные шаблоны. Пример шаблона для базового уровня: «Система классифицировала текст как [вердикт] с уверенностью [проценты]%. Основные причины: [список не более трех сильных аргументов]. Педагогическая рекомендация: [текст]».

3. Для детального уровня шаблон включает отдельные разделы с аргументами Защитника и Обвинителя с указанием силы каждого и итоговым перевесом.

4. Экспертный уровень генератора (L=2) предназначен для ситуаций, требующих глубокого анализа, используя большую языковую модель LLM. В отличие от базового и детального уровней, где применяются детерминированные шаблоны, здесь формируется объяснение, включающее интерпретацию признаков, методы измерения и статистические обоснования.

5. Сгенерированное объяснение E(L) возвращается Агенту-Судье, а также напрямую преподавателю.

На рисунке 1 представлена диаграмма потоков Модуля объяснимости.

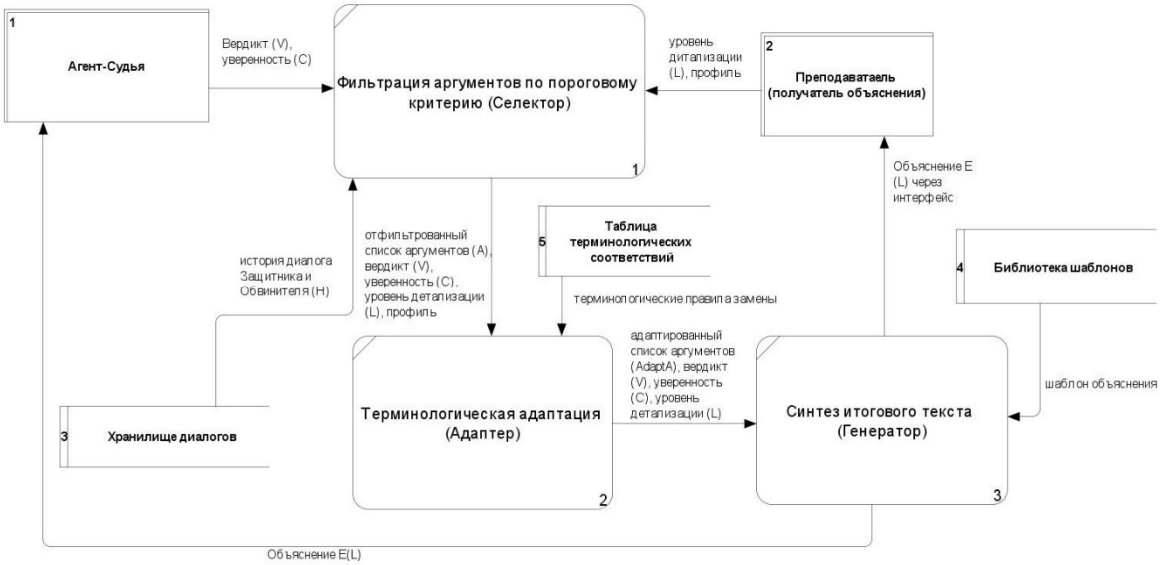

Рисунок 1 – Диаграмма потоков данных

Предложенное решение изначально проектирует объяснимость как неотъемлемое свойство процесса аргументации между специализированными

агентами.Языковая модель в модуле объяснимости может быть использована не только для генерации экспертных объяснений, но и для решения дополнительных задач. В АдаптереLLM способна динамически заменять термины, отсутствующие в статической таблице соответствия, с кэшированием результатов для повторного использования. В Генераторе языковая модель может применяться для формирования педагогических рекомендаций, учитывающих не только вердикт и уверенность, но и специфику конкретных аргументов.При этом любое применение языковой модели должно сопровождаться процедурой верификации выходных данных, включающей проверку числовых утверждений на соответствие исходной истории диалога между агентами и фильтрацию потенциально неверных формулировок.

Литература

1. Alexander K., Savvidou C., Alexander C. Who wrote this essay? Detecting AI-generated writing in second language education in higher education // Teaching English with Technology. 2023. Vol. 23. № 2. P. 25–43. DOI: https://doi.org/10.56297/BUKA4060/XHLD5365.

2. Мельников, Д. А., Петрова, А. А. Методы контроля использования искусственного интеллекта в образовании и их значение для развития учебного процесса / Д. А. Мельников, А. А. Петрова // Инновационное развитие техники и технологий в промышленности (ИНТЕКС-2025). Часть 5. — М.: ФГБОУ ВО «РГУ им. А.Н. Косыгина», 2025. — С. 22-25.

3. Осипенко, Л. Е. Текстовые генеративные нейросети в исследовательской деятельности студентов / Л. Е. Осипенко, А. В. Коротков // Мир науки, культуры, образования. – 2024. – № 4(107). – С. 90-93. – DOI 10.24412/1991-5497-2024-4107-90-93. – EDN COOBKW.

4. Хаджиева, Л. К. Методы распознавания вероятности применения систем искусственного интеллекта в текстовых документах / Л. К. Хаджиева, А. К. Чадаев // Экономика и управление: проблемы, решения. – 2025. – Т. 7, № 5(158). – С. 170-176. – DOI 10.36871/ek.up.p.r.2025.05.07.022. – EDN OEUTHL.