https://doi.org/10.17323/jle.2024.22244

Только неправильные ответы: генерация дистракторов для русскоязычных вопросов на понимание прочитанного текста с использованием переведенного набора данных

Логин Никита Вячеславович

Национальный исследовательский университет «Высшая школа экономики», г. Москва, Россия

АННОТАЦИЯ

Введение: Вопросы для проверки понимания прочитанного текста играют важную роль в изучении языка. Вопросы со множественным выбором (выбором ответа из нескольких вариантов) удобны для оценки понимания прочитанного текста, поскольку их можно проверять автоматически. Наличие больших датасетов (наборов данных) вопросов для проверки понимания прочитанного позволяет также автоматически создавать их с использованием тонкой настройки языковых моделей, снижая затраты на разработку тестовых банков вопросов. Датасеты для задачи понимания прочитанного текста на английском языке широко распространены, что нельзя сказать о подобных ресурсах для других языков, включая русский. Подзадача генерации дистракторов (неправильных вариантов ответа, англ. distractors) является особенно сложной, так как требует создания нескольких неправильных элементов.

Цель: Разработка эффективного решения для генерации дистракторов на русскоязычные вопросы, предназначенного для проверки понимания прочитанного текста в формате экзаменационных материалов. Также в этой работе ставится задача выяснить, может ли переведенный англоязычный датасет предложить возможность такого решения.

Методология: В данной работе мы тонко настроили (дообучили) две предварительно обученные большие русскоязычные модели RuT5 и RuGPT3 (Zmitrovich et al., 2024) для задачи генерации дистракторов к двум классам обобщающих вопросов. Эти вопросы были взяты из большого набора вопросов (автоматически переведенных с английского на русский) со множественным выбором. Первый класс составили вопросы, требующие выбора наиболее подходящего названия для данного отрывка, второй класс включал вопросы о выборе верных/неверных утверждений. Модели автоматически оценивались на основе тестовой подвыборки и подвыборки разработки, а модели дистракторов для вопросов с выбором истинных утверждений дополнительно оценивались на датасете вопросов ЕГЭ (Единого государственного экзамена).

Результаты: Тонко настроенные модели превзошли базовый вариант без тонкой настройки, производительность модели RuT5 оказалась лучше, чем у RuGPT3. Также было выявлено, что модели справлялись с вопросами о выборе истинных утверждений значительно лучше по сравнению с вопросами о выборе заголовка. На данных ЕГЭ модели, тонко настроенные на переведенном датасете, показали более высокое качество, чем тонко настроенные на существующем русскоязычном датасете. При этом модель на основе T5 также превзошла «бейзлайн» (базовый уровень качества), установленный выдачей существующей английской модели генерации дистракторов, переведенной на русский язык.

Вывод: Полученные результаты свидетельствуют о возможности использования переведенного набора данных для генерации дистракторов, а также подчеркивают важность совпадения предметной области (языкового экзамена) и типа вопроса в подвыборке для тонкой настройки и в новых примерах.

КЛЮЧЕВЫЕ СЛОВА

автоматическая генерация дистракторов, вопросы со множественным выбором, перевод прочитанного текста, большая языковая модель, перевод датасетов

Для цитирования: Логин, Н.В. (2024). Только неправильные ответы: Генерация отвлекающих факторов для вопросов на понимание прочитанного на русском языке с использованием переведенного набора данных. *Journal of Language and Education, 10*(4), 58-73. https://doi.org/10.17323/jle.2024.22244

Контакты:
Никита Логин
nlogin@hse.ru

Получена: 15 августа 2024

Принята: 16 декабря 2024

Опубликована: 30 декабря 2024

ВВЕДЕНИЕ

Автоматическая генерация вопросов является перспективной областью применения методов автоматической обработки естественного языка, так как она может многократно улучшить образовательный процесс. Согласно исследованию (Kurdi et al., 2020), стандартизированный экзамен обычно требует от организаторов экзаменов наличия больших банков вручную составляемых тестовых упражнений, которые должны регулярно обновляться для предотвращения списывания. Благодаря автоматизированной генерации эти банки могут пополняться непрерывно, что обеспечивает вариативность тестовых упражнений и снижает затраты на организацию экзаменов. Кроме того, автоматическая генерация упражнений может помочь тестируемым, поскольку она может предоставить им практически бесконечный источник тестовых заданий для подготовки.

Наличие достаточно качественных данных для обучения имеет решающее значение для автоматической генерации вопросов. Большинство наборов данных, используемых для обучения моделей в сфере автоматической генерации вопросов, изначально были разработаны для решения проблемы машинного понимания прочитанного, к ним относятся RACE (Lai et al., 2017), SciQ (Welbl et al., 2017), SQuAD (Rajpukar et al., 2016), COQA (Reddy et al., 2019), Natural Questions (Kwiatkowski et al., 2019) и TriviaQA (Joshi et al., 2017). Большинство этих датасетов включают в себя элементы, состоящие из фрагмента текста для чтения, набора вопросов, сопровождающих текст, правильного ответа и (необязательно) набора дистракторов для каждого вопроса. Тем не менее, существуют наборы данных, разработанные специально для генерации вопросов, в том числе QGSTEC (Rus et al., 2012) и FairyTaleQA (Xu et al., 2022). Среди упомянутых наборов датасетов наиболее примечательным является RACE, так как содержит вопросы экзаменационного формата, поскольку входящие в него задания изначально были извлечены из китайских веб-сайтов, содержащих экзаменационные материалы по английскому языку. Для русского языка следует отметить такие вопросные датасеты, как DaNetQA (Glushkova et al., 2021), MuSeRC/RuCoS (Fenogenova et al., 2020), SberQUAD (Efimov et al., 2020) и RuBQ (Rybin et al., 2021). DaNetQA и SberQuAD содержат вопросы на понимание параграфов из Википедии, полученные при помощи краудсорсинга, RuBQ основан на викторинах и материалах ресурса Wikidata, в MuSeRC и RuCoS включены вопросы, соответствующие текстовым абзацам, извлеченным из различных источников, эти вопросы получены при помощи краудсорсинга. Среди русскоязычных датасетов можно выделить MuSeRC, так как он является единственным датасетом, содержащим дистракторы.

Генерация дистракторов — это особо важная подзадача автоматической генерации вопросов. Преимущество включения дистракторов в материалы онлайн-тестирования заключается в создании возможности немедленной автоматизированной оценки тестов, что исключает вероятность несправедливого суждения (в отличие от вопросов с ответом в свободной форме). Однако эта подзадача остается одной из самых сложных по следующим причинам:

(1) При генерации дистракторов несколько выходов (разные независимые дистракторы) соответствуют одному входу.
(2) Невозможно составить исчерпывающий набор «заведомо подходящих» дистракторов для данного вопроса, что затрудняет оценку производительности обученной модели.
(3) Генерируемые выходные данные должны быть неверными в контексте заданного вопроса, но правильными с точки зрения языка (Kurdi et al., 2020, p. 145), а также не должны быть слишком нерелевантными по отношению к вопросу.

В связи с быстрым развитием нейронных сетей в 2020–2024 гг., наиболее популярным в настоящее время является подход к генерации вопросов на основе нейронных сетей. В основном это реализуется одним из трех способов:

(1) Путем обучения/тонкой настройки модели «последовательность — последовательность» (Seq2Seq), как в (Lee et al., 2020; Makhnytkina et al., 2020; Xiao et al., 2020; Xu et al., 2022; Hadifar et al., 2022; Manakul et al., 2023; Zhang et al., 2023).
(2) Путем тонкой настройки авторегрессионной (предназначенной для продолжения текста) большой языковой модели (Belyanova et al., 2022)
(3) Путем промптинга (получения ответа при помощи детального описания задачи на естественном языке) большой инструктированной или чат-модели (Elkins et al., 2023; Wang et al., 2023).

Для генерации дистракторов в основном используются те же методы: Seq2Seq (Qiu et al., 2020; Hadifar et al., 2022; DeFitero-Dominguez et al., 2024; Ghanem & Fyshe, 2024), авторегрессионные (Chung et al., 2020; Ghanem & Fyshe, 2024) и инструктированные (Bitew et al., 2023; Maity et al., 2024) модели.

Генерация вопросов и дистракторов обычно оценивается с использованием метрик, изначально разработанных для оценки машинного перевода и суммаризации текста, таких как BLEU (Papineni et al., 2002), METEOR (Banerjee and Lavie, 2005) и ROUGE (Lin, 2004). BLEU основана на геометрическом среднем значений модифицированной точности N-грамм. Модифицированная точность N-грамм рассчитывается как отношение числа слов, одновременно присутствующих в сгенерированной и эталонной последовательности к числу уникальных слов в последней. Максимальный порядок N-грамм, задействованных при вычислении BLEU, используется в качестве индикатора конкретного варианта этой метрики (BLEU-1, BLEU-2, …). Метрика ROUGE может быть основана на полноте, точности или их гармоническом среднем (F-мере) с равными весами и имеет варианты,

зависимые от совпадения N-грамм (ROUGE-N), и от наиболее длинной общей подпоследовательности (ROUGE-L). METEOR был разработан для решения выявленных проблем BLEU (отсутствие полноты и шумность анализируемых N-грамм) и основывается на F-мере совпадения униграмм с большим весом полноты по сравнению с точностью.

Тонкая настройка моделей «последовательность — последовательность» остается наиболее популярным решением проблемы генерации текстов вопросов. Lee et al. (2020) реализовали модель генерации вопросов на основе BiLSTM, которая обучалась совместно двум задачам — прогнозированию правильного ответа и прогнозированию текста вопроса. Xiao et al. (2020) обучили пользовательскую модель Multi-Flow Attention Transformer (Vaswani et al., 2017) на задаче прогнозирования текста вопроса с использованием набора данных SQuAD. Xu et al. (2022) тонко настроили модель BART (Lewis et al., 2020) на данных FairyTaleQA для генерации текстов вопросов и достигли оценки ROUGE-L F1 52,7. Hadifar et al. (2022) тонко настроили модель T5 (Raffel et al., 2020) для задачи генерации текстов вопросов на основе данных EduQG и SQuAD, достигнув оценок BLEU-4, METEOR и ROUGE-L 15,41; 29,65 и 34,26 соответственно. Wang et al. (2023) использовали для генерации текста вопроса решение на основе GPT-2 (Radford et al., 2019) без тонкой настройки, задействовав расширение техники Beam Search под названием NeuroLogicDecoding (Lu et al., 2021). Метод оценивался на наборе данных ClariQ-FKw (Sekulić et al., 2021), были получены оценки BLEU-4, ROUGE-L и METEOR 21,61; 41,03 и 47,87 соответственно.

Что касается генерации текстов вопросов для русскоязычных данных, то Makhnytkina et al. (2020) использовали модель Encode-Decoder на основе BiLSTM, обученную на разговорном наборе данных CoQA, автоматически переведенном на русский язык с помощью сервиса Яндекс.Переводчик. Модель достигла оценки BLEU-2 12,0. Belyanova et al. (2022) использовали модель RuGPT3, тонко настроенную на корпусах DaNetQA и RuBQ. Генерация проводилась авторегрессионным способом, текст вопроса прогнозировался как продолжение входной последовательности, текст правильного ответа не использовался. Модель достигла значений BLEU-4 4,75 и 1,95 на датасетах RuBQ и DaNetQA соответственно.

В генерации дистракторов также популярен подход «последовательность — последовательность». Qiu et al. (2020) использовали модель Seq2Seq, состоящую из энкодера на основе механизма внимания и декодера на основе BiLSTM, для тонкой настройки на дистракторах набора данных RACE-DG, специально предобработанной версии RACE и предназначенной для задачи генерации дистракторов (Gao et al., 2019). Они использовали параллельное декодирование в форме алгоритма Beam Search поверх предсказанных моделью распределений вероятностей слов, чтобы получить несколько дистракторов из одного входа, используя меру Жаккара для получения различающихся вариантов. Их модель показала баллы BLEU-4 7,57; 6,27; 5,27 для каждого из трех вариантов дистрактора соответственно. Chung et al. (2020) тонко настроили авторегрессионную языковую модель BERT (Devlin et al., 2019), используя совместное обучения по двум задачам: последовательному и параллельному предсказанию каждой лексемы дистрактора. Они использовали ту же схему непересекающейся генерации, что и Qiu et al. (2020), но применили энтропийный критерий вместо меры Жаккара. Для обучения и оценки использовался набор данных RACE-DG, баллы BLEU-4 и ROUGE-L на тестовой подвыборке составили 13,56 и 34,01 соответственно. Однако позже они выпустили[1] усовершенствованные версии своих моделей, основанные на архитектуре BART, которые достигли максимальных значений BLEU-4/ROUGE-L в 16,33 и 37,5 соответственно.

В более поздних работах по генерации дистракторов широко используется архитектура T5. Hadifar et al. (2022) реализовали генерацию дистракторов с помощью модели T5, обученной как на RACE, так и на собственном новом наборе данных EduQG. Весь набор дистракторов для каждого вопроса предсказывался сразу. Полученные баллы BLEU, METEOR, ROUGE-L на EduQG составили 17,73; 21,54; 34,13 соответственно. Ghanem & Fyshe (2024) точно настроили модели GPT-2 и T5 для задачи генерации дистракторов в рамках работы над метрикой качества генерации дистракторов DISTO, основанной на прогнозировании. Они использовали набор данных RACE для тонкой настройки и оценки и реализовали две версии T5: с совместной и параллельной генерацией дистракторов. Их лучшее решение, непересекающаяся модель T5, достигла 2,3 балла BLEU-4, в то время как GPT-2 и совместный T5 достигли всего 0,3 и 0,9 баллов BLEU-4 соответственно. De-Fitero-Dominguez et al. (2024) реализовали генерацию дистракторов с использованием модели mT5 (Xue et al., 2020), многоязычной версии T5, на комбинированном переведенном датасете дистракторов. Этот набор данных включал элементы из RACE-DG, CosmosQA (Huang et al., 2019) и SciQ, переведенные при помощи модели Opus-MT (Tiedemann & Thottingal, 2020). Их реализация достигла 7,21 и 21,76 по метрикам BLEU-4 и ROUGE-L на тестовой подвыборке RACE-DG.

Генерация дистракторов также рассматривалась как проблема ранжирования, как это было реализовано в работе Bitew et al. (2022). Модели были обучены выбирать наиболее подходящие дистракторы для данного вопроса и правильного ответа на него. Было реализовано два решения для ранжирования: одно с использованием инжиниринга признаков и логистической регрессии, и другое — на основе многоязычной модели BERT. Были задействованы три модели на основе BERT: первая была основана на сходстве дистрактора и правильного ответа, вторая — на сходстве дистрактора и текста вопроса, а третья, совместная модель,

[1] https://github.com/voidful/BDG

которая объединила две ранее упомянутые. В роли показателей качества использовались средняя точность и полнота ранжирования, самые высокие баллы (57,3 и 62,8 соответственно) были получены совместной моделью на основе BERT.

В самых последних работах также предпринимались эксперименты с подходом к генерации дистракторов на основе промптинга. Bitew et al. (2023) рассмотрели генерацию дистракторов с помощью модели T5, обученной на наборе данных Televic, и промптинга ChatGPT в конфигурациях zero-shot (с использованием промпта без примеров дистракторов) и one-shot (с использованием промпта с примерами). Все модели оценивались экспертами вручную. Maity et al. (2024) использовали многоуровневый «конвейер» на основе больших моделей ChatGPT и DaVinci, состоящий из генерации парафраза входного текста, извлечения ключевых слов из парафраза, генерации вопросов и дистракторов. Наилучшие значения BLEU-4 и ROUGE-L (2,49 и 13,54 соответственно) были получены с помощью многоуровневой модели, основанной на DaVinci.

Из всех рассмотренных работ только Qiu et al. (2020) специально затрагивают проблему потенциальной тривиальности дистракторов при проектировании своего решения. В своем исследовании они рассматривают тривиальность как нерелевантность данного вопроса представленному текстовому отрывку и утверждают, что эта проблема решается путем включения блоков, объединяющих информацию из отрывка для чтения и текста вопроса (называемых «Модулем реформулировки отрывка» и «Модулем реформулировки вопроса») в их исходную модель, основанную на Трансформере. При этом не делается однозначных выводов о том, как исключение обоих модулей реформулировки влияет на метрики (анализируется только исключение каждого из модулей в отдельности) и не учитываются ситуации, когда тривиальность не связана с отношением к входным данным.

Несмотря на то, что существует множество решений для генерации заданий на понимание прочитанного для английского языка, лишь небольшое количество таких решений было разработано для русского языка (Makhnytkina et al., 2020; Belyanova et al., 2022), а также не было найдено свидетельств существования разработок, которые решали бы задачу генерации дистракторов для русскоязычных вопросов. Кроме того, только один из существующих русскоязычных наборов данных вопросов для проверки понимания прочитанного содержит дистракторы, и ни один из этих наборов данных не ориентирован на существующие языковые экзамены. Ещё одна проблема заключается в том, что такие параметры, как тип и структура вопросов, не использовались при генерации дистракторов в предыдущих работах, в то время как учет этих параметров при проектировании модели генерации дистракторов может потенциально облегчить задачу для нейросети. Важность этих параметров подтверждается выводами Xu et al. (2022), которые реализовали категоризацию вопросов в дизайне своего набора данных вопросов FairyTaleQA с использованием системы нарративных элементов и отношений, описанной в Paris & Paris (2003). Они обнаружили, что характер ответа может зависеть от повествовательной категории вопроса, что проиллюстрировано на примере вопросов, относящихся к типу «Чувство».

Принимая во внимание важность автоматической генерации тестов на понимание прочитанного в экзаменационном формате и отсутствие методов для генерации дистракторов на русскоязычных данных, целью данной статьи является разработка решения для автоматической генерации дистракторов для экзаменационных вопросов на понимание прочитанного на русском языке. Из-за отсутствия датасетов дистракторов в формате экзамена по русскому языку мы также решили изучить возможность использования переведенного англоязычного датасета для генерации русскоязычных дистракторов, как это было сделано Makhnytkina et al. (2020) для генерации вопросов и De-Fitero-Dominguez et al. (2024) для испаноязычных данных. Кроме того, мы намеревались исследовать перспективы тонкой настройки моделей генерации дистракторов на конкретных категориях вопросов. Мы ожидали, что богатый и тщательно подобранный англоязычный датасет послужит эффективным источником обучающих данных, а обучение на определенной категории вопросов позволит лучше переносить знание, усвоенное моделью генерации дистракторов, на стандартизированные экзаменационные вопросы. Мы сформулировали наши исследовательские вопросы следующим образом:

ИВ1. Можно ли эффективно тонко настроить модель генерации дистракторов на англоязычном датасете, который был автоматически переведен на русский язык?

ИВ2. Есть ли необходимость в специальном русскоязычном наборе данных вопросов для проверки понимания прочитанного со множественным выбором для эффективной генерации дистракторов экзаменационного типа, или для этой задачи достаточно существующего датасета MuSeRC, имеющего не экзаменационный формат?

ИВ3. Может ли тонкая настройка на определенном типе вопросов привести к повышению производительности модели генерации дистракторов на стандартизированных экзаменационных данных?

МЕТОДОЛОГИЯ

Дизайн исследования

В настоящем исследовании мы провели эксперименты по тонкой настройке больших языковых моделей для решения задачи по генерации дистракторов к вопросам, предназначенным для проверки понимания прочитанного. В

процессе работы использовались различные наборы данных. Основное внимание мы уделили тонкой настройке на переведенном англоязычном наборе данных. Для этой цели мы использовали RACE, так как он содержит вопросы со множественным выбором, предназначенные для оценки понимания прочитанного в формате языкового экзамена, а также широко применялся во многих работах по генерации дистракторов (Chung et al., 2020; Qiu et al., 2020; De-Fitero-Dominguez et al., 2024; Ghanem & Fyshe, 2024). Кроме того, мы включили исходно русскоязычный набор данных для оценки понимания прочитанного с несколькими вариантами ответов (MuSeRC), чтобы понять, существует ли необходимость перевода иноязычного набора данных и можно ли достичь достаточного качества генерации, используя уже имеющиеся русскоязычные данные.

В наших экспериментах мы тонко настроили две большие языковые модели, выложенные в открытый доступ (Zmitrovich et al., 2024) командой AI-Forever — RuGPT3 и RuT5, которые являются русскоязычными реализациями моделей GPT3 (Brown et al., 2020) и T5 (Raffel et al., 2020). RuGPT3 использует авторегрессионную генерацию текста и состоит только из блоков декодера Трансформера, в то время как RuT5 основана на подходе «последовательность — последовательность» и содержит как энкодер, так и декодер. Для оценки сгенерированного нами результата мы использовали общепринятые метрики качества генерации для задач типа «последовательность — последовательность» (см. подраздел «Оценка» в разделе «Методика»).

Для сравнения моделей, обученных на разных наборах данных, мы также использовали небольшой набор изначально русскоязычных данных Единого государственного экзамена (ЕГЭ), полученных из открытых источников в сети Интернет. Мы также включили в нашу оценку «бейзлайны», в том числе базовую версию RuGPT3 без тонкой настройки, а также расширенные версии моделей от Chung et al. (2020), результаты работы которых были переведены на русский язык автоматически.

Наборы данных

RACE

RACE — это набор данных, состоящий из 98 000 вопросов на английском языке для проверки понимания прочитанного в формате китайского национального экзамена для учащихся средних и старших классов школы. Каждый текст RACE сопровождался несколькими вопросами со множественным выбором, для которых предлагалось по пять вариантов ответов: четыре правильных и один неправильный. Мы перевели набор данных RACE с помощью Opus-MT, модели перевода с английского на русский, доступной из пакета EasyNMT[2] для языка программирования Python. Каждый вопрос и набор дистракторов переводились в конкатенации с текстом для чтения, чтобы модель перевода не потеряла контекст. После описательного анализа вопросов RACE, выполненного в средах Microsoft Excel и Python, мы обнаружили две явно выделяющиеся категории вопросов, подходящие для генерации дистракторов:

- Вопросы, предлагающие участнику тестирования выбрать лучший заголовок для данного отрывка (TITLE);
- Вопросы, в которых участнику тестирования предлагается выбрать ИСТИННОЕ или ЛОЖНОЕ утверждение из заданного набора (TF).

Мы собрали вопросы этих категорий с помощью поиска по регулярным выражениям. Полученные с помощью этой методики наборы данных — Ru-RACE-TITLE и Ru-RACE-TF — содержали 4892 и 3799 элементов соответственно.

Для Ru-RACE-TITLE мы отобрали 805 уникальных формулировок вопроса из RACE, которые соответствовали регулярному выражению ***\Wtitle\W*** (т. е. содержали слово ***title***). Затем мы вручную отфильтровали 53 нерелевантные формулировки вопроса (например, содержащие слово *title* в значении «титул, социальный статус» или уточняющих название какого-либо упомянутого в тексте предмета). Полученный набор данных был разделен на подмножества train/test/dev с использованием исходных меток подмножеств из RACE, что привело к разделению 4575/219/242.

Для Ru-RACE-TF мы выбрали формулировки вопроса, которые при приведении к нижнему регистру соответствовали регулярному выражению ***which of the following .+(true|false).*** Таким образом было получено 693 уникальных формулировки вопроса. После ручной фильтрации были удалены 143 формулировки вопросов. Применив ту же логику разбиения, что и для Ru-RACE-TITLE, мы получили разделение 3288/175/187. Формулировка задания в Ru-RACE-TF идентична Заданию 18 ЕГЭ по русскому языку, что позволило нам использовать данные в формате ЕГЭ, описанные в разделе «ЕГЭ-TF» настоящей работы.

MuSeRC

Для сравнения производительности на переведенном наборе данных с производительностью на исходно русскоязычных данных мы также использовали MuSeRC. MuSeRC — это набор данных, созданный Fenogenova et al., (2020) в рамках бенчмарка RussianSuperGLUE. Он содержит 12 805 русскоязычных вопросов со множественным выбором для проверки понимания прочитанного, составленных краудсорсинговыми работниками из текстов различных предметных областей. Каждый текст сопровождается набором вопросов, при этом вопросы содержат, как правило, 1–2 правильных ответа и 2–3 дистрактора.

[2] https://github.com/UKPLab/EasyNMT

ЕГЭ-TF

Единый государственный экзамен — обязательный российский государственный экзамен, который используется для оценки знаний выпускников старшей школы и в качестве вступительного теста в высшие учебные заведения. ЕГЭ по русскому языку включает Задание 18, которое представляет собой вопрос для проверки понимания прочитанного с несколькими вариантами ответа, участникам предлагается выбрать ВЕРНЫЕ либо НЕВЕРНЫЕ утверждения из заданного набора. Данные для этого задания содержат тестовые задания, собранные Shavrina et al. (2020), а также полученные из других общедоступных Интернет-источников. В набор входило 55 уникальных вопросов, каждый из которых имел по 5 вариантов ответов. Некоторые вопросы содержали более одного правильного варианта, поэтому мы предварительно обработали их, как описано в разделе «Предварительная обработка данных» настоящей статьи.

Методика

Предварительная обработка данных

В RU-RACE-TITLE все тексты вопросов были заменены на фразу: «*Какое название лучше всего подойдет для этого текста?*». В RU-RACE-TF тексты вопросов были заменены либо на фразу: «*Какое высказывание СООТВЕТСТВУЕТ тексту?*», либо на фразу: «*Какое высказывание НЕ СООТВЕТСТВУЕТ тексту?*». Вопросы и варианты ответов MuSeRC были оставлены без изменений, так как мы хотели провести обучение и оценку на всех элементах соответствующих подвыборок данного набора данных.

Для ЕГЭ-TF мы применили ту же процедуру предобработки, что и для RuRACE, с внесением изменений, связанных с наличием нескольких правильных вариантов. Для заданий, в которых было больше правильных вариантов, чем неправильных, мы изменили текст вопроса на противоположный: «*Какое высказывание СООТВЕТСТВУЕТ тексту?*» было изменено на «*Какое высказывание НЕ СООТВЕТСТВУЕТ тексту?*» и наоборот. В этом случае первый по порядку из первоначальных дистракторов использовался в качестве правильного ответа, а исходные правильные варианты использовались в качестве дистракторов. Если же неправильных вариантов было больше, чем правильных, вопрос оставался неизменным, а в качестве правильного ответа использовался первый правильный вариант, в то время как дистракторы использовались без изменений.

Обучение моделей

Модели обучались на удаленном приватном сервере с графическим процессором Nvidia Tesla V100. Все модели обучались в течение 20 эпох с оптимизатором ADAM, используя начальную скорость обучения 5e-5 и коэффициент затухания весов 0,01. Мы определили максимальную длину выходных данных генерации как 0,99 квантиля длины примера обучающей подвыборки. Обучающие подвыборки использовались для тонкой настройки моделей, в то время как тестовые подвыборки и подвыборки разработки использовались для оценки.

На этапе обучения мы построили входные примеры для RuGPT3 в виде конкатенаций отрывка для чтения, текста вопроса, правильного ответа и набора дистракторов, разделенных знаками переноса строки, перемежая их русскими фразами, указывающими на составные части входного примера (*ВОПРОС*, *ПРАВИЛЬНЫЙ ОТВЕТ* и *НЕПРАВИЛЬНЫЕ ВАРИАНТЫ ОТВЕТА*). Для обучения RuT5 мы построили отдельные входные и выходные примеры, так как модель (в отличие от RuGPT3) работала не по принципу авторегрессии, а по принципу «последовательность — последовательность». Входной пример для RuT5 включал отрывок для чтения, вопрос и правильный ответ, разделенные теми же фразами, что и примеры RuGPT3, в то время как пример на выходе состоял из дистракторов, заключенных в двойные кавычки и разделенных точкой с запятой.

Генерация

На этапе генерации входные примеры для RuGPT3 имели ту же структуру, что и на этапе обучения, но включали только отрывок для чтения, текст вопроса и правильный ответ. Для моделей этой архитектуры мы генерировали текст до тех пор, пока не была достигнута установленная нами максимальная длина. Предсказанное продолжение входного примера разбивалось по строкам, после чего отфильтровывались дистракторы, которые были либо неуникальными, либо идентичными правильному ответу. Далее оставшиеся дистракторы сортировались в алфавитном порядке, после чего сохранялись первые три результата. Для генерации при помощи RuT5 мы использовали как максимально установленную длину, так и токен конца последовательности в качестве точки остановки генерации. Дистракторы извлекались при помощи разбиения вывода по точкам с запятой, при этом удалялись кавычки, в которые были заключены варианты.

Оценка

Для автоматической оценки сгенерированных дистракторов использовались значения метрик BLEU и METEOR. Использовались реализации BLEU и METEOR, доступные из пакета Evaluate[3] для языка программирования Python. Для того, чтобы наши результаты были сопоставимы с предыдущими и предстоящими работами, мы также включили метрику ROUGE-L в нашу оценку. Поскольку официальная реализация ROUGE, доступная из пакета Evaluate, несовме-

[3] https://pypi.org/project/evaluate/

стима с русскоязычными данными, мы использовали её неофициальную реализацию[4]. Однако её авторы признают, что полученные значения могут отличаться от официального варианта. Так как большинство предыдущих работ (Chung et al., 2020; Qiu et al., 2020; Belyanova et al., 2022; Wang et al., 2023; Maity et al., 2024) использовали 4-граммный вариант BLEU, эта конфигурация BLEU была использована по умолчанию и в нашей статье. Мы также использовали BERTScore (Zhang et al., 2023) для семантической оценки сгенерированных дистракторов. BERTScore — это метрика, основанная на сходстве эмбеддингов слов, полученных при помощи модели BERT, а не на точных совпадениях слов/N-грамм. Для удобства интерпретации результатов все значения метрик (определенные от 0 до 1) были представлены в процентах, в диапазоне от 0 до 100.

«Бейзлайны»

Поскольку во время работы над этой статьей мы сформировали набор данных ЕГЭ-TF и внесли оригинальные модификации в набор данных RACE, было решено не ограничиваться сообщением результатов из предыдущих работ, а проверить модели «бейзлайнов» на наших данных. Это было сделано для того, чтобы обеспечить справедливое сравнение, так как метрики, использованные в нашей оценке, нельзя напрямую сравнивать с данными разных языков. Реализация моделей «бейзлайнов» описана в настоящем разделе, а результаты представлены и проанализированы в разделах «Результаты» и «Обсуждение».

В качестве стартовой точки в наших экспериментах мы использовали базовую версию RuGPT3 без тонкой настройки. Вместе с RuGPT3 в конфигурации «нулевого выстрела» (zero-shot) мы также использовали модели BART-DG — улучшенные версии моделей, представленных Chung et al. (2020). В настоящее время у них наиболее высокие результаты с точки зрения BLEU при генерации дистракторов на данных RACE. Для получения русскоязычного вывода от этих моделей использовалась та же процедура автоматического перевода, что и при создании Ru-RACE. Входные данные ЕГЭ перед передачей в модели BART-DG переводились на английский язык с применением той же многоязычной модели перевода (Opus-MT).

РЕЗУЛЬТАТЫ

Ru-RACE

В Таблице 1 показаны результаты обучения моделей на переведенных подвыборках RACE. Для обеих задач наилучшие результаты показали модели на базе T5 (RuT5-RACE-TITLE и RuT5-RACE-TF). Это можно объяснить тем, что T5 работает по принципу «последовательность — последовательность», что позволяет ей преобразовывать входы в выходы, имеющие отличную от входов структуру. Обе модели превзошли «бейзлайн», установленный для обеих задач RuGPT3 без тонкой настройки, из чего можно сделать вывод, что тонкая настройка позволила им успешно адаптироваться к структуре наших переведенных наборов данных.

Для Ru-RACE-TITLE наивысшее качество с точки зрения BLEU, METEOR и BERTScore как на подвыборке разработки, так и на тестовой подвыборке было достигнуто за счет тонко настроенной модели RuT5. Тонко настроенная RuGPT3 показала определенно более низкие результаты, при этом BLEU-4 достиг всего 3,83 и 3,19 для подвыборок разработки и тестирования соответственно (по сравнению с 25,17 и 22,96 у RuT5-RACE-TITLE). Более заметная разница наблюдается в значениях метрики METEOR: 12,78/12,41 на наборах для разработки/тестирования для RuGPT3 против 46,09/45,35 для RuT5. Эти различия говорят о том, что модель RuT5-RACE-TITLE значительно превосходит RuGPT3-RACE-TITLE как по точности, так и по полноте. Что касается BERTScore, то разница между двумя тонко настроенными моделями на тестовом наборе (10,04) выше, чем разница между моделью с наименьшей оценкой и базовой моделью (4,24), что указывает на то, что способность RuT5-RACE-TITLE создавать

Таблица 1

Результаты моделирования на трансляционных подмножествах RACE

	BLEU-4		METEOR		ROUGE-L		BERTScore	
	dev	**test**	**dev**	**test**	**dev**	**test**	**dev**	**test**
Ru-RACE-TITLE								
RuGPT3-RACE-TITLE	3.83	3.19	12.78	12.41	12.32	12.60	68.72	68.68
RuT5-RACE-TITLE	**25.17**	**22.96**	**46.09**	**45.35**	**16.79**	**16.21**	**79.09**	**78.72**
Baseline RuGPT3	0.46	0.53	5.37	5.57	4.31	4.47	62.72	62.46
Ru-RACE-TF								
RuGPT3-RACE-TF	8.75	4.89	18.92	16.84	16.16	13.80	71.01	70.23
RuT5-RACE-TF	**26.36**	**22.43**	**44.84**	**42.75**	**28.36**	**25.30**	**77.07**	**76.24**
Baseline RuGPT3	1.23	1.73	9.54	9.44	8.30	8.29	63.64	64.04

[4] https://github.com/pltrdy/rouge

семантически согласованные дистракторы значительно превосходит таковую у RuGPT3-RACE-TITLE. С точки зрения значений ROUGE-L две тонко настроенные модели не так уж далеки друг от друга (4,47/3,61 на подмножествах для разработки/тестирования), но они обе значительно превосходят «бейзлайн». Значения качества генерации всех моделей на подвыборках для разработки и тестирования довольно близки друг к другу, что подтверждает отсутствие переобучения моделей, т. е. они не адаптировались слишком сильно к данным тестовых подвыборок во время настройки гиперпараметров.

Для Ru-RACE-TF наивысшее качество по BLEU-4/METEOR/BERTScore также показала тонко настроенная модель RuT5, в то время как качество генерации тонко настроенной RuGPT3 было значительно ниже (на 17,54/25,91/6,01 балла на тестовой подвыборке соответственно). Значения BLEU-4 и METEOR модели RuGPT3-RACE-TITLE расположены ближе к базовому уровню, чем к значениям RuT5-RACE-TITLE. Тем не менее различия по BERTScore между второй по качеству моделью и «бейзлайном», а также между первой и второй по качеству моделями составляют 7,37/6,19 и 6,06/6,01 соответственно (на подвыборках разработки/тестирования). Это говорит о том, что разрыв по семантической согласованности между двумя тонко настроенными моделями может быть не таким уж большим. Разница по ROUGE-L достаточно велика как между второй по качеству моделью и «бейзлайном», так и между первой и второй по качеству моделями. При этом качество генерации дистракторов между подвыборками разработки и тестирования для Ru-RACE-TF отличается несущественно, что свидетельствует об отсутствии переобучения. Можно заметить, что все оценки моделей для этой задачи выше, чем в Ru-RACE-TITLE.

MuSeRC

В Таблице 2 показаны результаты обучения моделей на наборе данных MuSeRC. Поскольку тестовая подвыборка MuSeRC была недоступна на веб-сайте разработчика набора данных, все оценки была произведены на подвыборке разработки. Обе модели превзошли базовый уровень, установленный GPT3 без дополнительного обучения, по всем трем показателям. Значение метрики BLEU-4 у RuT5-MuSeRC-DG почти в два раза выше, чем у RuGPT3-MuSeRC-DG (23,62 против 12,48), а значение метрики METEOR у RuT5-MuSeRC-DG лишь немного выше (45,78 против 40,87). Согласно значениям BERTScore (76,04 и 76,02 для RuT5-MuSeRC-DG и RuGPT3-MuSeRC-DG соответственно), дистракторы, генерируемые этими двумя моделями, почти одинаково семантически близки к дистракторам золотого стандарта. С точки зрения ROUGE-L, значения двух тонко настроенных моделей довольно близки, обе модели значительно превосходят базовый уровень «нулевого выстрела». Поскольку обе тонко настроенные модели дают результаты, которые превосходят базовый уровень без тонкой настройки, они были использованы для проверки на данных ЕГЭ.

ЕГЭ-TF

В Таблице 3 представлены результаты работы с набором данных ЕГЭ-TF. Наибольших значений метрик достигла модель RuT5-RACE-TF, второе место заняла модель BART-DG-PM. Тем не менее мы видим, что оценки переведенной выдачи BART-DG-PM (11,02/28,47/70,90 по BLEU-4/METEOR/BERTScore) довольно близки к оценкам нашей лучшей модели (11,64/29,61/71,06 соответственно).

Можно заметить, что модель RuT5-RACE-TF демонстрирует устойчивость при работе с данными ЕГЭ, так как её значения BLEU, METEOR и BERTScore все ещё значительно превышают базовый уровень без тонкой настройки. Однако это не относится к другим моделям, обученным на русскоязычных данных, поскольку их значения метрик значительно уступают и приближаются к базовым уровням, установленным RuGPT3 без тонкой настройки. Значения BLEU всех моделей, за исключением RuT5-RACE-TF и вариантов BART-DG, быстро снижаются до нуля с повышением ранга BLEU, что говорит о недостаточной устойчивости этих моделей. Особенно это касается моделей, обученных на MuSeRC, это означает, что существующие русскоязычные датасеты не могут предложить данные, пригодные для генерации дистракторов в сложных общих заданиях, направленных на проверку понимания прочитанного, которые встречаются в материалах языковых экзаменов. Можно сделать вывод, что обучение на переведенном наборе данных обеспечивает устойчивость результатов, в то время как обучение на существующем русскоязычном наборе данных — нет. Это можно объяснить тем, что набор данных MuSeRC содержит более тривиальные тексты, чем те, что можно найти в заданиях ЕГЭ, поскольку MuSeRC состоит в основном из новостных сообщений, в то время как тексты ЕГЭ, как правило, представляют собой отрывки из литературных произведений средней школы, поднимающие этические проблемы, заслуживающие обсуждения. В связи с тем, что RuT5-RACE-TF является нашей единственной устойчивой моделью для сравнения, в дальнейшем планируется использовать только её прогнозы при ручной оценке данных, полученных на основе ЕГЭ-TF.

ОБСУЖДЕНИЕ РЕЗУЛЬТАТОВ

Результаты (Таблицы 1–3) демонстрируют, что производительность на переведенных наборах данных для задачи генерации дистракторов находится на одном уровне с существующими работами, при этом значения BLEU-4 достигают максимума, около 25, для тестовых подвыборок наборов данных, на которых они были изначально тонко настроены. Тонко настроенные модели на основе RuT5 показали в целом лучшую производительность при создании согласованных дистракторов, чем модели на основе RuGPT3. Это может быть связано с тем, что RuT5 была предварительно обучена на задаче реконструкции, а не генерации текста, и, таким образом, более подвержена тонкой настройке. Из

Таблица 2

Результаты моделирования на датасете MuSeRC

	BLEU-4	METEOR	ROUGE-L	BERTScore
RuGPT3-MuSeRC-DG	12.48	40.87	21.77	**76.04**
RuT5-MuSeRC-DG	**23.62**	**45.78**	**25.97**	76.02
Baseline RuGPT3	5.16	11.25	6.81	62.91

Таблица 3

Результаты моделирования на наборе данных ЕГЭ-TF

	BLEU-1	BLEU-2	BLEU-3	BLEU-4	METEOR	ROUGE-L	BERTScore
RuGPT3-RACE-TF	15.11	3.57	0.08	0.00	9.22	6.83	65.84
RuT5-RACE-TF	**29.11**	**20.93**	**15.66**	**11.64**	**29.61**	**13.55**	**71.06**
RuGPT3-MuSeRC	9.56	1.69	0.48	0.00	6.65	4.35	61.62
RuT5-MuSeRC	10.30	2.20	0.55	0.00	7.77	4.66	62.63
Baseline RuGPT3	11.22	2.00	0.53	0.00	7.57	4.74	55.72
BART-DG	26.66	19.44	14.64	10.78	27.77	12.57	70.83
BART-DG-PM	28.52	20.24	15.06	11.02	28.47	12.40	70.90
BART-DG-ANPM	27.39	19.71	14.64	10.71	27.78	11.75	70.62

наших результатов видно, что только RuT5 смогла выдать когерентные выходные данные как на своем исходном наборе данных, так и на независимом наборе вопросов ЕГЭ, в то время как другие наши тонко настроенные модели смогли сделать это только на данных из тестовой подвыборки набора данных, на котором они были тонко настроены. В этом разделе мы объясним взаимосвязи между оценками различных моделей на разных наборах данных и предложим способы улучшения наших результатов, сравнивая их с результатами предыдущих исследований по этой теме.

Неспособность моделей, обученных на MuSeRC, выдавать когерентные выходные данные дистракторов для ЕГЭ можно объяснить характером набора данных MuSeRC и относительно низкой сложностью его элементов по сравнению с реальными заданиями на проверку понимания прочитанного. Тот факт, что BART-DG (расширенные версии моделей из Chung et al., 2020) удерживает сильный «бейзлайн» по сравнению с нашими результатами на данных ЕГЭ, можно объяснить сложной структурой моделей BDG, которая включает в себя дополнительные методы инжиниринга, добавленные к базовой модели. Эти методы включают в себя декодирование на основе максимизации энтропии на разных путях генерации для создания нескольких дистракторов независимо друг от друга (в то время как наши модели генерируют их последовательно), параллельное многозадачное обучение (PM) и ответ-негативную регуляризацию (AN).

В нашем исследовании показано, что модели на основе T5 более эффективно тонко настраиваются на дистракторах из вопросов RACE по сравнению с моделями других архитектур подтверждается работами Hadifar et al. (2022), Ghanem & Fyshe (2024) и De-Fitero-Dominguez et al. (2024). В работе Ghanem & Fyshe (2024) также обнаружен разрыв в метриках качества генерации между моделями, состоящими только из декодера и моделями с энкодером-декодером. В частности, тонко настроенные модели T5-base значительно превосходят тонко настроенные GPT2-small (8,4 и 13,7 с точки зрения BLEU-2 для совместной и параллельной T5-генерации соответственно, тогда как показывает только 3,9).

Результаты сравнения с BDG на том же наборе данных отличаются от выводов Ghanem & Fyshe (2024) Авторы этой работы обнаружили, что их лучшая модель превосходит BDG только с точки зрения BLEU-1 (32,0 против 30,2). Однако стоит отметить, что они тонко настроили модель BDG на своих данных, тогда как в нашей работе мы использовали уже готовые тонко настроенные варианты BDG. Кроме того, они реализовали параллельную генерацию дистракторов (как это также было сделано Chung et al., 2020), что позволило повысить производительность, в то время как в нашей работе реализован только совместный подход.

Мы обнаружили, что качество генерации на переведенном наборе данных более высокое, чем на исходном языке, аналогичный вывод был сделан в исследовании De-Fitero-Dominguez et al. (2024). Однако это могло быть вызвано сокращением лексического пространства, обусловленным объемом словарного запаса модели перевода. При этом результаты на переведенных данных никогда не были особенно высокими. Так, в работе Makhnytkina et al. (2020) модель генерации вопросов продемонстрировала низкую производительность по формальным метрикам на тестовом подмножестве набора данных, на котором она была тонко настроена. Тем не менее эти результаты можно объяснить

использованием старой модельной архитектуры (BiLSTM) и меньшим развитием моделей перевода в 2020 г.

Результаты наших экспериментов, в которых оценка METEOR во всех условиях превышает показатели ROUGE-L, противоречат выводам Hadifar et al. (2022), где оценка METEOR была значительно ниже ROUGE-L. Принимая во внимание различия в расчете метрик (гораздо больший вес полноты в METEOR, чем в ROUGE, использование наибольшей общей длины подпоследовательности в ROUGE-L вместо совпадения униграмм в METEOR), можно сделать вывод, что их модель лучше усваивает паттерны из данных дистракторов, но менее эффективно сохраняет лексическое содержание. В то время как для наших решений верно обратное.

Учитывая недостаточную производительность моделей на основе GPT3 в наших экспериментах, стоит отметить, что для тонкой настройки мы использовали «маленькую» версию RuGPT3 из-за нехватки вычислительных ресурсов, необходимых для тонкой настройки «больших» версий RuGPT3. С учетом близости наших результатов на независимом датасете ЕГЭ-TF к моделям BART-DG, следует подчеркнуть, что в них реализовано параллельное декодирование выходных дистракторов, в то время как в наших решениях используется совместное декодирование. Принимая во внимание преимущество в значениях метрик параллельной генерации по сравнению с совместной, обнаруженное Ghanem & Fyshe (2024), можно предположить, что при параллельном декодировании наши модели смогут значительно превзойти BART-DG.

Наши предположения относительно результатов, описанных в разделе «Введение» (эффективность тонкой настройки на переведенном высококачественном наборе данных и преимущество тонкой настройки на конкретном типе вопросов), подтверждаются результатами модели RuT5-RACE-TF на данных ЕГЭ-TF, которая демонстрирует производительность, превосходящую как «бейзлайны», так и модели, обученные на русскоязычном наборе данных с вопросами разных типов. Однако модель RuGPT3-RACE-TF не соответствует нашим предположениям: она показывает лучшие результаты по сравнению с моделями, обученными на MuSeRC, но ее показатели не достигают уровня «бейзлайнов», установленных моделями, обученными на полном наборе данных RACE-DG.

Важно отметить, что представленные формальные количественные метрики основаны на сходстве между сгенерированными и изначальными дистракторами, а общая уместность дистракторов требует оценки человеком, как описано в подразделе «Будущая работа». В наших моделях используются базовые реализации T5 и GPT3, но в дальнейшем возможна их оптимизация посредством инженерных усовершенствований.

Дальнейшие направления исследования

Для дальнейшей ручной оценки наших моделей, тонко настроенных на Ru-RACE-TF, планируется использовать данные ЕГЭ-TF (составленный профессионалами набор исходно русскоязычных вопросов с несколькими вариантами ответов) и включать прогнозы моделей RuT5-RACE-TF и BART-DG-PM. Каждый вопрос ЕГЭ-TF планируется снабдить 4 вариантами ответа: исходным правильным ответом, дистрактором-филлером, одним из исходных дистракторов, одним прогнозом нашей тонко настроенной модели и одним прогнозом BART-DG-PM. Дистрактором-филлером может быть предложение, извлеченное из существующего корпуса русского языка, которое семантически близко (что подтверждается формальными метриками, такими как BERTScore) к отрывку из читаемого текста. Планируется привлечь русскоязычных участников с высшим образованием, чтобы тестируемые действительно могли отличать дистракторы от настоящих правильных ответов. Участникам будет предложено оценить каждый из примеров по шкале от 1 до 5, где 1 будет указывать на самый неподходящий вариант, а 5 — на вариант, который с наибольшей вероятностью будет правильным ответом. Гипотеза состоит в том, что дистракторы из наших моделей будут в среднем оценены выше, чем дистракторы-филлеры, но ниже, чем исходные правильные ответы.

Для ручной оценки нашей лучшей модели, обученной на Ru-RACE-TITLE (RuT5-RACE-TITLE), планируется использовать произвольные русскоязычные тексты в качестве входных данных для моделей генерации дистракторов. Это могут быть отрывки из газетных источников и художественные рассказы для детей, так как большинство текстов из RACE носят повествовательный характер. Дизайн вопросов будет таким же, как и для проверочного подмножества Ru-RACE-TF, правильный ответ будет представлять собой оригинальное название статьи либо будет создаваться вручную. Гипотетически ожидаются те же выводы о взаимосвязях между средними значениями рангов, что и в датасете ЕГЭ-TF. Поскольку значения метрик для тонко настроенных моделей на Ru-RACE-TF были выше, чем на Ru-RACE-TITLE, можно ожидать, что при предлагаемой ручной оценке средние ранги дистракторов для Ru-RACE-TF также будут выше, чем ранги для Ru-RACE-TITLE.

Оценивая общую релевантность сгенерированных вариантов, этот метод также позволяет определить частоту появления тривиальных дистракторов, поскольку ожидается, что тривиальные варианты будут в среднем оценены наравне с дистракторами-филлерами. Кроме того, целесообразно провести аннотирование набора прогнозов дистракторов модели с точки зрения релевантности и тривиальности. Аннотация характеристик дистракторов, полученных в ходе этой процедуры, может быть использована для будущего обучения ML-модели, предназначенной для оценки дис-

тракторов. Несмотря на то, что тривиальность в последнее время обычно не рассматривается отдельно в аналогичных работах, поскольку современные генеративные нейронные сети способны выявлять закономерности из представленных неструктурированных данных без необходимости дополнительного инженерной обработки выдачи, эта модель может быть использована во время будущего обучения нового конвейера генерации дистракторов. В частности, она позволит «наказывать» за выходы, которые будут определены как слишком тривиальные, и вознаграждать за выходы, оценка уместности которых будет высокой. Дополнительные усовершенствования могут включать в себя параллельное декодирование выходных данных, использование более крупных вариантов моделей и реализацию подхода на основе промптинга в качестве альтернативы.

ЗАКЛЮЧЕНИЕ

В данной работе была реализована автоматическая генерация дистракторов для русскоязычных данных. Для задачи генерации дистракторов были тонко настроены шесть больших языковых моделей двух типов (GPT-3 и T5) на трех наборах данных: двух наборах вопросов, автоматически переведенных с английского на русский, содержащих только определенные типы вопросов (выбор названия и выбор верных/неверных утверждений), а также одного изначально русскоязычного набора. Модели на базе RuT5 показали в целом лучшие результаты, чем модели на базе RuGPT3. Оба типа моделей превзошли «бейзлайн», установленный моделью RuGPT3 без тонкой настройки, что доказывает возможность эффективной тонкой настройки для генерации дистракторов на данных, переведенных с английского на русский язык. Этот аспект касается ИВ1.

В ходе экспериментов с русскоязычными экзаменационными данными было установлено, что переведенный на английский язык тестовый набор более эффективен с точки зрения использования в тонкой настройке моделей, чем существующий русскоязычный не экзаменационный набор данных, поскольку модели, обученные на последнем наборе данных, показали низкую производительность по сравнению с моделями, обученными на первом. Это подчеркивает важность предметной области и уровня сложности вопросов для задачи генерации дистракторов и доказывает необходимость наличия комплексного набора вопросов для проверки понимания прочитанного с несколькими вариантами ответов в формате российского экзамена. Таким образом, получаем ответ на ИВ2.

Модель на основе T5, тонко настроенная на дистракторах для вопросов с выбором верных утверждений, продемонстрировала лучшую производительность при работе с данными ЕГЭ по сравнению с моделями, обученными на MuSeRC, а также с передовым решением для генерации дистракторов в экзаменационном формате. Это подтверждает преимущество тонкой настройки на определенном типе вопросов, что было предметом рассмотрения в ИВ3.

Основная ценность работы заключается в обучении моделей генерации дистракторов для русскоязычных данных, что ранее не было сделано. Особый интерес представляет исследование возможности переноса знаний, полученных на конкретных категориях вопросов из больших наборов данных, на генерацию дистракторов для экзаменационных вопросов определенного стандарта. Наши результаты могут быть полезны при создании платформ для подготовки к экзаменам. Разработчики могут включить модели, обученные описанным в настоящей работе методом, в свои продукты, что позволит автоматически пополнять банки заданий.

Наши выводы также могут заинтересовать создателей наборов данных для задачи понимания прочитанного. Они могут включить аннотацию различных общих типов вопросов в дизайн своих наборов данных. Отсутствие русскоязычных наборов данных, необходимых для успешного обучения моделей, может предоставить им возможность для заполнения пустующей ниши.

Дальнейшую работу в этом направлении целесообразно сосредоточить на проведении ручной оценки сгенерированных данных, а также на разработке русскоязычного набора данных для проверки понимания прочитанного в формате экзаменационных заданий. Кроме того, перспективным представляется изучение возможностей различных альтернативных методов генерации, которые не были рассмотрены в наших экспериментах. Ещё одним интересным направлением для будущих исследований может стать детальное сравнение русскоязычных дистракторов, полученных из моделей, обученных для их генерации, с дистракторами, извлеченными из больших инструктированных и чат-моделей,

БЛАГОДАРНОСТИ

Автор выражает благодарность своему научному руководителю — профессору Ольге Николаевне Ляшевской, за предоставление ценных консультаций при подготовке доработанной рукописи. Исследование осуществлено в рамках Программы фундаментальных исследований НИУ ВШЭ.

КОНФЛИКТ ИНТЕРЕСОВ

Автор заявляет об отсутствии конфликта интересов.

ЛИТЕРАТУРА

Alsubait, T. M. (2015). *Ontology-based multiple-choice question generation* [Unpublished PhD thesis]. University of Manchester.

Banerjee, S., & Lavie, A. (2005). METEOR: An automatic metric for MT evaluation with improved correlation with human judgements. In J. Goldstein, A. Lavie, C.-Y. Lin, & C. Voss (Eds.), *Proceedings of the ACL Workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization* (pp. 65–72). Association for Computational Linguistics.

Belyanova, M. A., Andreev, A. M., & Gapanyuk, Y. E. (2022). Neural text question generation for Russian language using hybrid intelligent information systems approach. In B. Kryzhanovsky, W. Dunin-Barkowski, V. Redko, Y. Tiumentsev, & V. V. Klimov (Eds.), *Advances in neural computation, machine learning, and cognitive research V* (vol. 1008, pp. 217–223). Springer International Publishing. http://dx.doi.org/10.1007/978-3-030-91581-0_29

Bitew, S. K., Hadifar, A., Sterckx, L., Deleu, J., Develder, & C., Demeester, T. (2022) Learning to reuse distractors to support multiple choice question generation in education. *IEEE Transactions on Learning Technologies, 17,* 375–390. IEEE Computer Society Press. http://dx.doi.org/10.1109/TLT.2022.3226523

Bitew, S. K., Deleu, J., Develder, C., & Demeester, T. (2023) *Distractor generation for multiple-choice questions with predictive prompting and large language models* (Version 1). arXiv. http://dx.doi.org/10.48550/arXiv.2307.16338

Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D. M., Wu, J., Winter, C., Hesse, C., Chen, M., Sigler, E., Litwin, M., Gray, S., Chess, B., Clark, J., Berner, C., McCandlish, S., Radford, A., Sutskever, I., & Amodei, D. (2020). Language models are few-shot learners. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, & H. Lin (Eds.), *Advances in Neural Information Processing Systems* (vol. 33, pp. 1877–1901). Curran Associates, Inc. http://dx.doi.org/10.48550/arXiv.2005.14165

Chung, H.-L., Chan, Y.-H., & Fan, Y.-C. (2020). A BERT-based distractor generation scheme with multi-tasking and negative answer training strategies. In T. Cohn, Y. He, & Y. Liu (Eds.), *Findings of the Association for Computational Linguistics: EMNLP 2020* (pp. 4390–4400). Association for Computational Linguistics. http://dx.doi.org/10.18653/v1/2020.findings-emnlp.393

De-Fitero-Dominguez, D., Garcia-Lopez, E., Garcia-Cabot, A., Del-Hoyo-Gabaldon, J.-A., & Moreno-Cediel, A. (2024). Distractor generation through text-to-text transformer models. *IEEE Access*, *12*, 25580–25589. http://dx.doi.org/10.1109/ACCESS.2024.3361673

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. In J. Burstein, C. Doran, & T. Solorio (Eds.), *Proceedings of the 2019 conference of the North American Chapter of the Association for Computational Linguistics: Human language technologies* (vol. 1: Long and Short Paper, pp. 4171–4186). Association for Computational Linguistics. http://dx.doi.org/10.18653/v1/N19-1423

Efimov, P., Chertok, A., Boytsov, L., & Braslavski, P. (2020). SberQuAD – Russian reading comprehension dataset: Description and analysis. In A. Arampatzis, E. Kanoulas, T. Tsikrika, S. Vrochidis, H. Joho, C. Lioma, C. Eickhoff, A. Névéol, L. Cappellato, & N. Ferro (Eds.), *Experimental IR meets multilinguality, multimodality, and interaction* (vol. 12260, pp. 3–15). Springer International Publishing. http://dx.doi.org/10.1007/978-3-030-58219-7_1

Elkins, S., Kochmar, E., Serban, I., & Cheung, J. C. K. (2023). How useful are educational questions generated by large language models? In N. Wang, G. Rebolledo-Mendez, V. Dimitrova, N. Matsuda, & O. C. Santos (Eds.), *Artificial intelligence in education. Posters and late breaking results, workshops and tutorials, industry and innovation tracks, practitioners, doctoral consortium and blue sky* (vol. 1831, pp. 536–542). Springer Nature Switzerland. http://dx.doi.org/10.1007/978-3-031-36336-8_83

Fenogenova, A., Mikhailov, V., & Shevelev, D. (2020). Read and reason with MuSeRC and RuCoS: Datasets for machine reading comprehension for Russian. In D. Scott, N. Bel, & C. Zong (Eds.), *Proceedings of the 28th International Conference on Computational Linguistics* (pp. 6481–6497). International Committee on Computational Linguistics. http://dx.doi.org/10.18653/v1/2020.coling-main.570

Gao, Y., Bing, L., Li, P., King, I., & Lyu, M. R. (2019). Generating distractors for reading comprehension questions from real examinations. *Proceedings of the AAAI Conference on Artificial Intelligence*, *33*(01), 6423–6430. http://dx.doi.org/10.1609/aaai.v33i01.33016423

Ghanem, B. & Fyshe, A. (2024). DISTO: Textual distractors for multiple choice reading comprehension questions using negative sampling. In M. Marras, M. Ueno (Eds.), *Proceedings of the 17th International Conference on Educational Data Mining* (pp. 23–34). International Educational Data Mining Society. http://dx.doi.org/10.5281/ZENODO.12729766

Glushkova, T., Machnev, A., Fenogenova, A., Shavrina, T., Artemova, E., & Ignatov, D. I. (2021). DaNetQA: A yes/no question answering dataset for the Russian language. In W. M. P. Van Der Aalst, V. Batagelj, D. I. Ignatov, M. Khachay, O. Koltsova, A. Kutuzov, S. O. Kuznetsov, I. A. Lomazova, N. Loukachevitch, A. Napoli, A. Panchenko, P. M. Pardalos, M. Pelillo, A. V. Savchenko, & E. Tutubalina (Eds.), *Analysis of Images, Social Networks and Texts* (vol. 12602, pp. 57–68). Springer International Publishing. http://dx.doi.org/10.1007/978-3-030-72610-2_4

Hadifar, A., Bitew, S. K., Deleu, J., Develder, C., & Demeester, T. (2023). EduQG: A multi-format multiple-choice dataset for the educational domain. *IEEE Access, 11*, 20885–20896. http://dx.doi.org/10.1109/ACCESS.2023.3248790

Huang, L., Le Bras, R., Bhagavatula, C., & Choi, Y. (2019). CosmosQA: Machine reading comprehension with contextual commonsense reasoning. In K. Inui, J. Jiang, V. Ng, & X. Wan (Eds.), *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)* (pp. 2391–2401). Association for Computational Linguistics. http://dx.doi.org/10.18653/v1/D19-1243

Joshi, M., Choi, E., Weld, D., & Zettlemoyer, L. (2017). TriviaQA: A large scale distantly supervised challenge dataset for reading comprehension. In R Barzilay., & M.-Y. Kan (Eds.), *Proceedings of the 55th Annual Meeting of the Association for Computational linguistics* (vol. 1: Long Papers, pp. 1601–1611). Association for Computational linguistics. http://dx.doi.org/10.18653/v1/P17-1147

Kurdi, G., Leo, J., Parsia, B., Sattler, U., & Al-Emari, S. (2020). A systematic review of automatic question generation for educational purposes. *International Journal of Artificial Intelligence in Education*, *30*(1), 121–204. http://dx.doi.org/10.1007/s40593-019-00186-y

Kwiatkowski, T., Palomaki, J., Redfield, O., Collins, M., Parikh, A., Alberti, C., Epstein, D., Polosukhin, I., Devlin, J., Lee, K., Toutanova, K., Jones, L., Kelcey, M., Chang, M.-W., Dai, A. M., Uszkoreit, J., Le, Q., & Petrov, S. (2019). Natural questions: A benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, *7*, 453–466. http://dx.doi.org/10.1162/tacl_a_00276

Lai, G., Xie, Q., Liu, H., Yang, Y., & Hovy, E. (2017). RACE: Large-scale reading comprehension dataset from examinations. In M. Palmer, R. Hwa, & S. Riedel (Eds.), *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing* (pp. 785–794). Association for Computational Linguistics. http://dx.doi.org/10.18653/v1/D17-1082

Lee, D. B., Lee, S., Jeong, W. T., Kim, D., & Hwang, S. J. (2020). Generating diverse and consistent QA pairs from contexts with information-maximizing hierarchical conditional VAEs. In D. Jurafsky, J. Chai, N. Schluter, & J. Tetreault (Eds.), *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics* (pp. 208–224). Association for Computational Linguistics. http://dx.doi.org/10.18653/v1/2020.acl-main.20

Lewis, M., Liu, Y., Goyal, N., Ghazvininejad, M., Mohamed, A., Levy, O., Stoyanov, V., & Zettlemoyer, L. (2020). BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In D. Jurafsky, J. Chai, N. Schluter, & J. Tetreault (Eds.), *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics* (pp. 7871–7880). Association for Computational Linguistics. http://dx.doi.org/10.18653/v1/2020.acl-main.703

Lin, C.-Y. (2004). ROUGE: A Package for automatic evaluation of summaries. In *Text summarization branches out* (pp. 74–81). Association for Computational Linguistics. https://aclanthology.org/W04-1013

Lu, X., West, P., Zellers, R., Bras, R. L., Bhagavatula, C., & Choi, Y. (2021). NeuroLogic decoding: (Un)supervised neural text generation with predicate logic constraints. In K. Toutanova, A. Rumshisky, L. Zettlemoyer, D. Hakkani-Tur, I. Beltagy, S. Bethard, R. Cotterell, T. Chakraborty, & Y. Zhou (Eds.), *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human language technologies* (pp. 4288–4299). Association for Computational Linguistics. http://dx.doi.org/10.18653/v1/2021.naacl-main.339

Maity, S., Deroy, A., & Sarkar, S. (2024). A novel multi-stage prompting approach for language agnostic MCQ generation using GPT. In N. Goharian, N. Tonellotto, Y. He, A. Lipani, G. McDonald, C. Macdonald, & I. Ounis (Eds.), *Advances in information retrieval* (vol. 14610, pp. 268–277). Springer Nature Switzerland. http://dx.doi.org/10.1007/978-3-031-56063-7_18

Makhnytkina, O., Matveev, A., Svischev, A., Korobova, P., Zubok, D., Mamaev, N., & Tchirkovskii, A. (2020). Conversational question generation in Russian. In S. Balandin, L. Turchet, & T. Tyutina (Eds.), *2020 27th Conference of Open Innovations Association (FRUCT)* (pp. 1–8). IEEE. http://dx.doi.org/10.23919/FRUCT49677.2020.9211056

Manakul, P., Liusie, A., & Gales, M. (2023). MQAG: Multiple-choice question answering and generation for assessing information consistency in summarization. In J. C. Park, Y. Arase, B. Hu, W. Lu, D. Wijaya, A. Purwarianti, & A. A. Krisnadhi (Eds.), *Proceedings of the 13th International Joint Conference on Natural Language Processing and the 3rd Conference of the Asia-Pacific chapter of the Association for Computational Linguistics* (vol. 1: Long Papers, pp. 39–53). Association for Computational Linguistics. http://dx.doi.org/10.18653/v1/2023.ijcnlp-main.4

Papineni, K., Roukos, S., Ward, T., & Zhu, W.-J. (2002). BLEU: a method for automatic evaluation of machine translation. In *The 40th Annual Meeting on Association for Computational Linguistics—ACL '02* (pp. 311–318). Association for Computational Linguistics. http://dx.doi.org/10.3115/1073083.1073135

Paris, A. H., & Paris, S. G. (2003). Assessing narrative comprehension in young children. *Reading Research Quarterly*, *38*(1), 36–76. http://dx.doi.org/10.1598/RRQ.38.1.3

Qiu, Z., Wu, X., & Fan, W. (2020). Automatic distractor generation for multiple choice questions in standard tests. In D. Scott, N. Bel, & C. Zong (Eds.), *Proceedings of the 28th International Conference on Computational Linguistics* (pp. 2096–2106). International Committee on Computational Linguistics. http://dx.doi.org/10.18653/v1/2020.coling-main.189

Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., & Liu, P. J. (2020). Exploring the limits of transfer learning with a unified text-to-text transformer. *The Journal of Machine Learning Research, 21, 1*, 5485–5551. https://dl.acm.org/doi/abs/10.5555/3455716.3455856

Rajpurkar, P., Zhang, J., Lopyrev, K., & Liang, P. (2016). SQuAD: 100,000+ questions for machine comprehension of text. In J. Su, K. Duh, & X. Carreras (Eds.), *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing* (pp. 2383–2392). Association for Computational Linguistics. http://dx.doi.org/10.18653/v1/D16-1264

Reddy, S., Chen, D., & Manning, C. D. (2019). CoQA: A conversational question answering challenge. *Transactions of the Association for Computational Linguistics, 7*, 249–266. http://dx.doi.org/10.1162/tacl_a_00266

Rybin, I., Korablinov, V., Efimov, P., & Braslavski, P. (2021). RuBQ 2.0: An innovated Russian question answering dataset. In R. Verborgh, K. Hose, H. Paulheim, P.-A. Champin, M. Maleshkova, O. Corcho, P. Ristoski, & M. Alam (Eds.), *The Semantic Web* (vol. 12731, pp. 532–547). Springer International Publishing. http://dx.doi.org/10.1007/978-3-030-77385-4_32

Sekulić, I., Aliannejadi, M., & Crestani, F. (2021). Towards facet-driven generation of clarifying questions for conversational search. In *Proceedings of the 2021 ACM SIGIR International Conference on Theory of Information Retrieval* (pp. 167–175). Association for Computing Machinery. http://dx.doi.org/10.1145/3471158.3472257

Shavrina, T., Emelyanov, A., Fenogenova, A., Fomin, V., Mikhailov, V., Evlampiev, A., Malykh, V., Larin, V., Natekin, A., Vatulin, A., Romov, P., Anastasiev, D., Zinov, N., & Chertok, A. (2020, May). Humans keep it one hundred: An overview of AI Journey. In N. Calzolari, F. Béchet, P. Blache, K. Choukri, C. Cieri, T. Declerck, S. Goggi, H. Isahara, B. Maegaard, J. Mariani, H. Mazo, A. Moreno, J. Odijk, & S. Piperidis (Eds.), *Proceedings of the Twelfth Language Resources and Evaluation Conference* (pp. 2276–2284). European Language Resources Association. https://aclanthology.org/2020.lrec-1.277/

Tiedemann, J., & Thottingal, S. (2020). OPUS-MT – Building open translation services for the world. In A. Martins, H. Moniz, S. Fumega, B. Martins, F. Batista, L. Coheur, C. Parra, I. Trancoso, M. Turchi, A. Bisazza, J. Moorkens, A. Guerberof, M. Nurminen, L. Marg, & M. L. Forcada (Eds.), *Proceedings of the 22nd Annual Conference of the European Association for Machine Translation* (pp. 479–480). European Association for Machine Translation. https://aclanthology.org/2020.eamt-1.61/

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł. ukasz, & Polosukhin, I. (2017). Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, & R. Garnett (Eds.), *Advances in Neural Information Processing Systems* (vol. 30, 6000–6010). Curran Associates, Inc. https://dl.acm.org/doi/10.5555/3295222.3295349

Welbl, J., Liu, N. F., & Gardner, M. (2017). Crowdsourcing multiple choice science questions. In L. Derczynski, W. Xu, A. Ritter, & T. Baldwin (Eds.), *Proceedings of the 3rd Workshop on Noisy User-generated Text* (pp. 94–106). Association for Computational Linguistics. http://dx.doi.org/10.18653/v1/W17-4413

Xiao, D., Zhang, H., Li, Y., Sun, Y., Tian, H., Wu, H., & Wang, H. (2020). ERNIE-GEN: An enhanced multi-flow pre-training and fine-tuning framework for natural language generation. In C. Bessiere (Ed.) *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence* (pp. 3997–4003). International Joint Conferences on Artificial Intelligence Organization. http://dx.doi.org/10.24963/ijcai.2020/553

Xu, Y., Wang, D., Yu, M., Ritchie, D., Yao, B., Wu, T., Zhang, Z., Li, T., Bradford, N., Sun, B., Hoang, T., Sang, Y., Hou, Y., Ma, X., Yang, D., Peng, N., Yu, Z., & Warschauer, M. (2022). Fantastic questions and where to find them: FairytaleQA – An authentic dataset for narrative comprehension. In S. Muresan, P. Nakov, & A. Villavicencio (Eds.), *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics* (vol. 1: Long Papers, pp. 447–460). Association for Computational Linguistics. http://dx.doi.org/10.18653/v1/2022.acl-long.34

Xue, L., Constant, N., Roberts, A., Kale, N., Al-Rfou, R., Siddhant, A., Barua, A., & Raffel, C. (2020). *MT5: A massively multilingual pre-trained text-to-text transformer* (Version 3). arXiv. http://dx.doi.org/10.48550/arXiv.2010.11934

Zhang, C. (2023). *Automatic generation of multiple-choice questions* (Version 1). arXiv. http://dx.doi.org/10.48550/ARXIV.2303.14576

Zhang, T., Kishore, V., Wu, F., Weinberger, K. Q., & Artzi, Y. (2019). *BERTScore: Evaluating text generation with BERT* (Version 3). arXiv. http://dx.doi.org/10.48550/ARXIV.1904.09675

Zmitrovich, D., Abramov, A., Kalmykov, A., Tikhonova, M., Taktasheva, E., Astafurov, D., Baushenko, M., Snegirev, A., Kadulin, V., Markov, S., Shavrina, T., Mikhailov, V., & Fenogenova, A. (2024). *A family of pretrained transformer language models for Russian*. In N. Calzolari, M.-Y. Kan, V. Hoste, A. Lenci, S. Sakti, & N. Xue (Eds.), *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)* (pp. 507–524). ELRA Language Resource Association. http://dx.doi.org/10.48550/arXiv.2309.10931

ПРИЛОЖЕНИЕ А

Пример генерации модели

Чтобы лучше проиллюстрировать работу нашей лучшей модели, RuT5-RACE-TF, мы проанализируем пример дистракторов, сгенерированных ей на произвольном русском тексте — детском рассказе Льва Толстого «Акула», взятом из русской Викитеки. Рассказ повествует нам о том, как корабельный артиллерист спас двух мальчиков, купавшихся в открытом море, от акулы, выстрелив в неё из пушки. Правильный ответ на вопрос «Какое высказывание СООТВЕТСТВУЕТ тексту?» («Меткость старого артиллериста спасла мальчиков от морского чудовища») был создан вручную.

Из Рисунка 1 видно, что варианты 2 и 3 действительно могут служить дистракторами, так как они противоречат тексту, поскольку в них излагаются факты, которых нет в рассказе («Старый артиллерист отвел мальчика в сторону»; «Мальчики, которые были в лодке, не слышали крика старого артиллериста»). Однако вариант 4 («История произошла в тот день, когда мы увидели акулу»), хоть и очень тривиален, но соответствует тексту и поэтому не может служить дистрактором в данном контексте. Следует отметить, что в приведенном примере язык сгенерированных дистракторов строен, последователен и не нарушает правила русской грамматики.

Наш корабль стоял на якоре у берега Африки. День был прекрасный, с моря дул свежий ветер; но к вечеру погода изменилась: стало душно и точно из топленной печки несло на нас горячим воздухом с пустыни Сахары. Перед закатом солнца капитан вышел на палубу, крикнул: «Купаться!» — и в одну минуту матросы попрыгали в воду, спустили в воду парус, привязали его и в парусе устроили купальню.
На корабле с нами было два мальчика. Мальчики первые попрыгали в воду, но им тесно было в парусе, они вздумали плавать наперегонки в открытом море.
Оба, как ящерицы, вытягивались в воде и что было силы поплыли к тому месту, где был бочонок над якорем.
Один мальчик сначала перегнал товарища, но потом стал отставать. Отец мальчика, старый артиллерист, стоял на палубе и любовался на своего сынишку. Когда сын стал отставать, отец крикнул ему: «Не выдавай! Понатужься!»
Вдруг с палубы кто-то крикнул: «Акула!» — и все мы увидали в воде спину морского чудовища.
Акула плыла прямо на мальчиков.
— Назад! Назад! Вернитесь! Акула! — закричал артиллерист. Но ребята не слыхали его, плыли дальше, смеялись и кричали еще веселее и громче прежнего.
Артиллерист, бледный как полотно, не шевелясь, смотрел на детей.
Матросы спустили лодку, бросились в нее и, сгибая весла, понеслись что было силы к мальчикам; но они были еще далеко от них, когда акула уже была не дальше 20-ти шагов.
Мальчики сначала не слыхали того, что им кричали, и не видали акулы; но потом один из них оглянулся, и мы все услыхали пронзительный визг, и мальчики поплыли в разные стороны.
Визг этот как будто разбудил артиллериста. Он сорвался с места и побежал к пушкам. Он повернул хобот, прилег к пушке, прицелился и взял фитиль.
Мы все, сколько нас ни было на корабле, замерли от страха и ждали, что будет.
Раздался выстрел, и мы увидали, что артиллерист упал подле пушки и закрыл лицо руками. Что сделалось с акулой и с мальчиками, мы не видали, потому что на минуту дым застлал нам глаза.
Но когда дым разошелся над водою, со всех сторон послышался сначала тихий ропот, потом ропот этот стал сильнее, и, наконец, со всех сторон раздался громкий, радостный крик.
Старый артиллерист открыл лицо, поднялся и посмотрел на море.
По волнам колыхалось желтое брюхо мертвой акулы. В несколько минут лодка подплыла к мальчикам и привезла их на корабль.

Какое высказывание СООТВЕТСТВУЕТ тексту?

1. **Меткость старого артиллериста спасла мальчиков от морского чудовища**
2. Старый артиллерист отвел мальчика в сторону
3. Мальчики, которые были в лодке, не слышали крика старого артиллериста
4. История произошла в день, когда мы увидели акулу

Примечание. Составленный вручную правильный ответ выделен жирным шрифтом.

ПРИЛОЖЕНИЕ B

Программный код и файлы данных для этой статьи доступны в онлайн-репозитории: https://github.com/nicklogin/Ru-RC-DG