Финальный проект

- **Литвинов Вячеслав**
- **Моисеев Даниил**

Постановка задачи и анализ данных

Задача

Даны тексты стихотворений с указанием жанра, ссылками на первоисточник, а также информацией о количестве просмотров и текущих пользовательских рейтингах. Все данные собраны с одного литературного сайта. Задача — создать систему, способную автоматически оценивать качество стихотворений, основываясь на доступной информации.

Ожидается, что модель будет достаточно универсальной, чтобы выполнять следующие задачи:

1. **Отбор данных для обучения** — выбор более качественных стихов, которые можно использовать для дальнейшего обучения.
2. **Ранжирование кандидатов** — оценка и упорядочивание N стихотворений-кандидатов во время инференса.
3. **Контрастивное ранжирование** — сравнение стихотворений для контрастивного обучения, что может улучшить восприятие модели к различиям в качестве.

Таким образом, модель должна принимать на вход текст стихотворения и возвращать оценку его качества, пригодную для ранжирования.

Данные

Контекст: Все стихи в датасете взяты с одного литературного сайта, ориентированного на современное сообщество авторов и читателей. Это произведения, созданные в определенном культурном контексте и отражающие вкусы и стили, принятые внутри этого сообщества. Из-за этого специфика данных может привести к тому, что модель будет переобучаться на стиль и систему оценок, характерные для этой аудитории, а не на объективные качества стихотворений. Поэтому важно учитывать, что модель, натренированная на этих данных, скорее всего, будет оценивать произведения с помощью подходов к

оценке, принятых в этом сообществе. Это ограничивает возможность ее применения для оценки других литературных произведений.

Ограничения: В данных присутствует жанр для каждого стиха, но для обучения модели эта информация использоваться не будет, так как по условиям задачи на входе должно быть только стихотворение.

Target

Для оценки стихотворений можно рассмотреть различные подходы к выбору целевой переменной:

- количество просмотров
- рейтинг
- комбинация этих двух показателей.

Однако существуют важные нюансы, которые могут повлиять на объективность модели:

- **Просмотры**. Высокое число просмотров может быть связано с удачным индексированием страницы в поисковых системах или популярностью известных произведений, загруженных на сайт. Это может привести к тому, что популярные стихи будут оцениваться выше, независимо от их качества.
- **Рейтинг**. Высокие показатели рейтинга могут отражать активную поддержку конкретных авторов их аудиторией, а не только объективные качества текста. Кроме того, система рейтингов сайта непрозрачна — неясно, по какой формуле она рассчитывается. Оценка может превышать количество просмотров, что говорит о том, что она является не просто "лайками".
- **Комбинирование просмотров и рейтинга**. Объединение просмотров и рейтинга в единую целевую переменную могло бы потенциально учесть оба аспекта, однако не ясно какую формулу использовать для их объединения.

Выбор метрики и обоснование

Для обучения модели регрессии (предсказание просмотров или оценок) будет использоваться MSE loss. Но, так как нам важен порядок предсказаний, для оценки точности предсказаний лучше подходят коэффициенты ранговой корреляции Кендалла и Спирмена. Оба коэффициента измеряют порядок между двумя списками, что делает их менее чувствительными к абсолютным значениям целевой переменной и полезными при сравнении моделей, с разными диапазонами выходных значений.

Эти коэффициенты принимают значения от -1 до +1, где +1 указывает на полное совпадение порядка предсказаний с истинным порядком, а 0 — на отсутствие какой-либо порядковой зависимости.

Оценщик для произвольного стихотворения

Использование CatBoost над текстовыми фичами

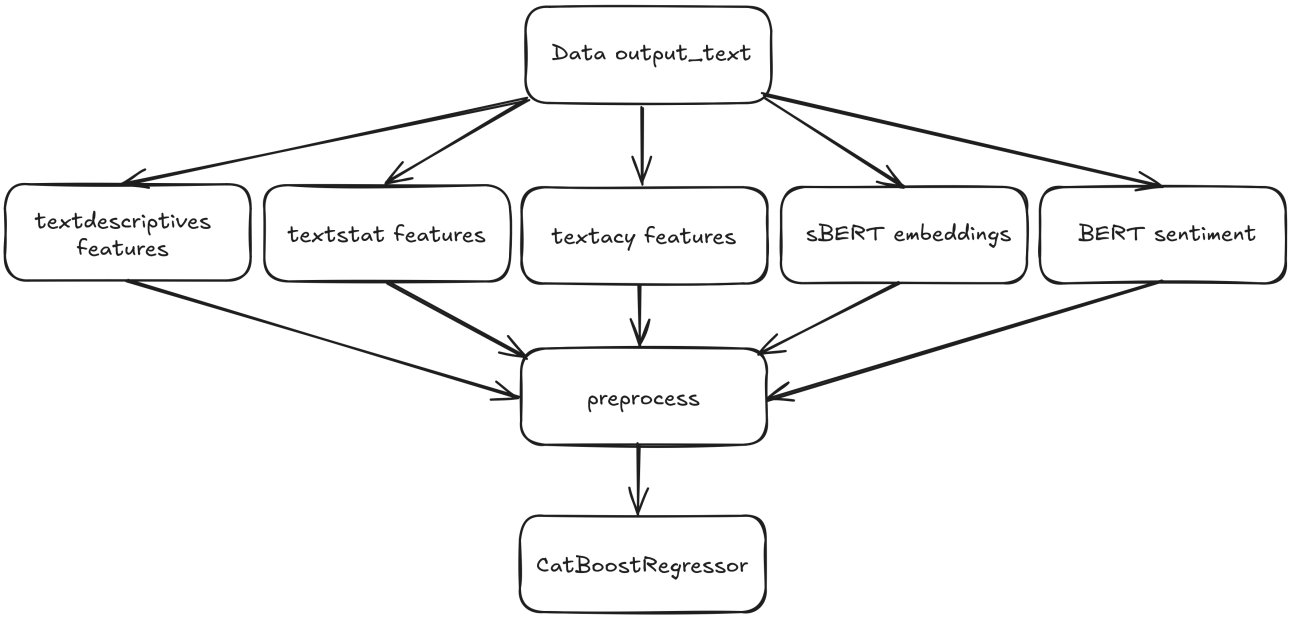
Кратко: Попробовали собрать множество текстовых фичей из различных библиотек, чтобы проверить, сможет ли модель хорошо предсказывать рейтинг стихотворений на их основе.

Мы предположили, что рейтинг стихотворения можно предсказать на основе разнообразных текстовых характеристик. Для вычисления этих характеристик использовали библиотеки TextDescriptives, textstat и textacy, которые помогли нам получить следующие показатели:

- Количество слов и предложений
- Метрики сложности текста
- Оценки читаемости и время необходимое на прочтение
- Метрики разнообразия слов
- Количество существительных, прилагательных и других частей речи
- И другие метрики.

Кроме того, были добавлены эмбеддинги из модели sBERT и оценки сентимента с помощью tiny-BERT модели. После очистки данных и удаления лишних признаков мы получили около 110 фичей + эмбеддинги sBERT.

Обучение модели CatBoost проводилось на размеченных данных, которые были разделены на обучающую и валидационную выборки в соотношении 80/20. Использовалась техника early stopping, чтобы предотвратить переобучение: обучение останавливалось, когда значение метрики коэффициента корреляции Кендалла на валидационной выборке переставало расти.



Важными характеристиками, выделенными моделью CatBoost, оказались как текстовые фичи, так и эмбеддинги. Наиболее значимые фичи включают:

Features	Importance
output_text	29.473032
textstat_difficult_words	2.356093
textacy_n_chars	1.588140
embedding_71	1.399563
textstat_letter_count	1.339400
embedding_141	1.219181
textdescriptives_top_ngram_chr_fraction_2	1.204721
embedding_633	1.100763
textstat_sentence_count	1.086004
sentiment	1.000711

Наиболее важной фичей оказался входной текст стихотворения, который модель обрабатывает с помощью метода Bag-of-Words. Кроме того, определенные метрики, отражающие сложность текста и сентимент, также играют значимую роль в предсказании рейтинга.

LLama3.2 с промптом для оценивания

Из-за отсутствия токенов для ChatGPT мы решили обратиться к опенсорсным LLM-моделям. Мы протестировали модели cyberlis/saiga-mistral:7b-lora-custom-q4_K и owl/t-lite от Т-банка, но обе продемонстрировали низкое качество работы. Они не справились с анализом стихотворений и показали слабые результаты в генерации ответов на русском языке.

В результате мы запустили локально модель Llama 3.2 и предоставили ей следующий промпт для оценки стихотворений:

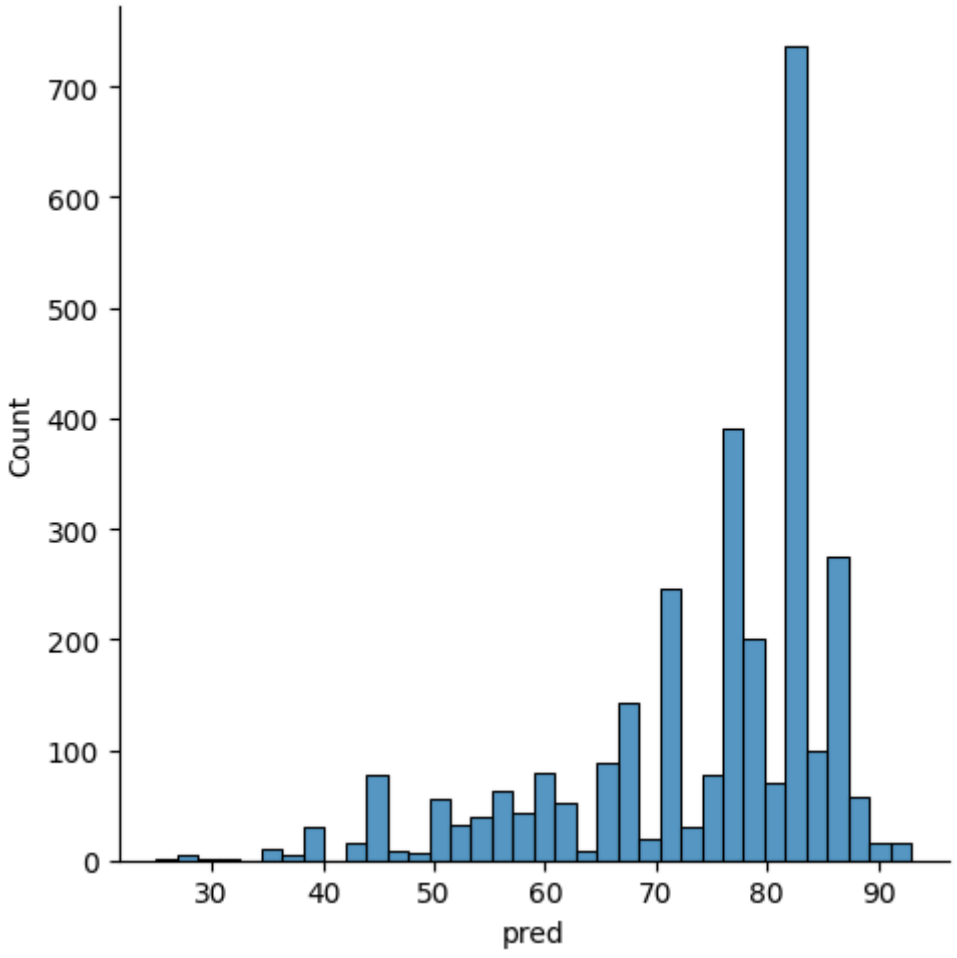
```
Проанализируй данное стихотворение и оцени его по следующим
критериям, выставляя итоговую оценку от 0 до 100, где 0 — низкое
качество, а 100 — высочайшее качество:

- Качество рифмы: оцени, насколько рифмы точны, гармоничны и уместны
(0–100).
- Смысл и глубина: оцени, насколько содержателен текст, передает ли
он глубокие идеи или оригинальные мысли, вызывает ли эмоции (0–100).
- Лексическое богатство и выразительность: оцени выбор слов, их
сочетаемость и выразительность (0–100).
- Ритм и структура: оцени плавность ритма и соблюдение формы и
структуры стихотворения (0–100).
- Общее впечатление: оцени общий эффект от стихотворения, насколько
оно звучит цельно и выразительно (0–100).

В конце напиши "Финальная оценка" и выведи итоговый балл от 0 до 100.
"""
{Текст стиха}
"""
```

Модель выдает структурированные оценки по каждому из указанных аспектов, а в конце формирует финальный балл, который используется как итоговый ответ. Однако в процессе тестирования мы заметили несколько важных особенностей работы Llama 3.2:

- **Склонность к высоким оценкам:** Модель редко выставляет низкие оценки, стремясь всегда находить значения около 80.
- **Вариативность результатов:** На одинаковые стихотворения модель может выдавать очень разные оценки при повторных запусках, что делает

результаты очень нестабильными.

На графике ниже представлено распределение ответов, полученных от Llama 3.2:



Результаты

Мы обучили модели CatBoost на трех различных таргетах: рейтинг (rating), просмотры (views) и комбинированная метрика, которая учитывает как рейтинг, так и количество просмотров. Для комбинированной метрики использовалась следующая формула:

$$\text{score} = \text{views} + e^{\left(1+\frac{\text{rating}}{\text{rating}+\text{views}}\right)} \cdot \text{rating}$$

Эта формула позволяет объединить просмотры и рейтинг стиха в одну метрику, учитывая отношение рейтинга к количеству просмотров. Например, для стиха с 150 просмотрами и нулевым рейтингом итоговый score составит 150, в то время как для стиха с 75 просмотрами и 75 рейтингом итоговый score будет равен 411.

Ниже представлены результаты оценки моделей на тестовом наборе данных с использованием коэффициентов корреляции Кендалла и Спирмена:

Model	Target	Kendall's Tau on test split	Spearman on test split
Catboost	Rating	0.2064	0.2783
Catboost	Views	0.1000	0.1488
Catboost	f(Rating,Views)	0.1115	0.1667
LLama3.2	Rating	0.1277	0.1665
LLama3.2	Views	0.0014	0.0017
LLama3.2	f(Rating, Views)	0.0398	0.0578

Выводы:

- **Лучшая модель:** Наилучшие результаты продемонстрировала модель CatBoost, обученная на таргете "рейтинг", с коэффициентами Кендалла и Спирмена равными 0.2064 и 0.2783 соответственно.
- **Комбинированная метрика:** Объединение просмотров и рейтинга в одну метрику показало плохие результаты.
- **Модель Llama3.2:** Результаты Llama3.2 значительно уступают модели CatBoost, особенно для таргета "просмотры", где коэффициенты близки к нулю. Также её результаты очень нестабильны.

Sanity check

Для проведения санити-чека мы оценили следующие стихи:

- **Евгений Онегин** А. С. Пушкина
- **Заметался пожар голубой** С. А. Есенина
- **Случайно сгенерированный стих** с помощью ChatGPT, содержащий хорошие рифмы, но без смысла
- **Лучший и худший стих** из тестовой выборки
- **Случайный список существительных**

Стих и автор	Выход модели LLama3.2	Выход CatBoost	Условный true ranking
Евгений Онегин (Пушкин)	82	7.8	10
Заметался пожар голубой (Есенин)	72	14	10
Бессмысленный стих из ChatGPT	82	7	3
Топ стих по рейтингу из теста	83	13	7
Стих с 0 просмотров из теста	51	4	4
Случайный набор существительных	45	3	0

Результаты метрик для sanity-check

Model	Metric	Value
Llama3.2	Kendall's Tau	0.3571
CatBoost	Kendall's Tau	0.6900
Llama3.2	Spearman	0.4558
CatBoost	Spearman	0.8406

Выводы:

- **Эффективность моделей:** CatBoost показал значительно лучшие результаты по обоим коэффициентам, чем Llama3.2, что подтверждает его более высокую точность в оценке качества стихотворений.
- **Непредсказуемые оценки:** Выходы Llama3.2 для бессмысленного стиха и известных произведений довольно высоки, что указывает на возможные ограничения модели в контексте понимания глубокого содержания.
- **Случайный набор существительных:** Оценка случайного набора существительных показывает, что обе модели не в состоянии распознать полное отсутствие поэтической структуры или содержания

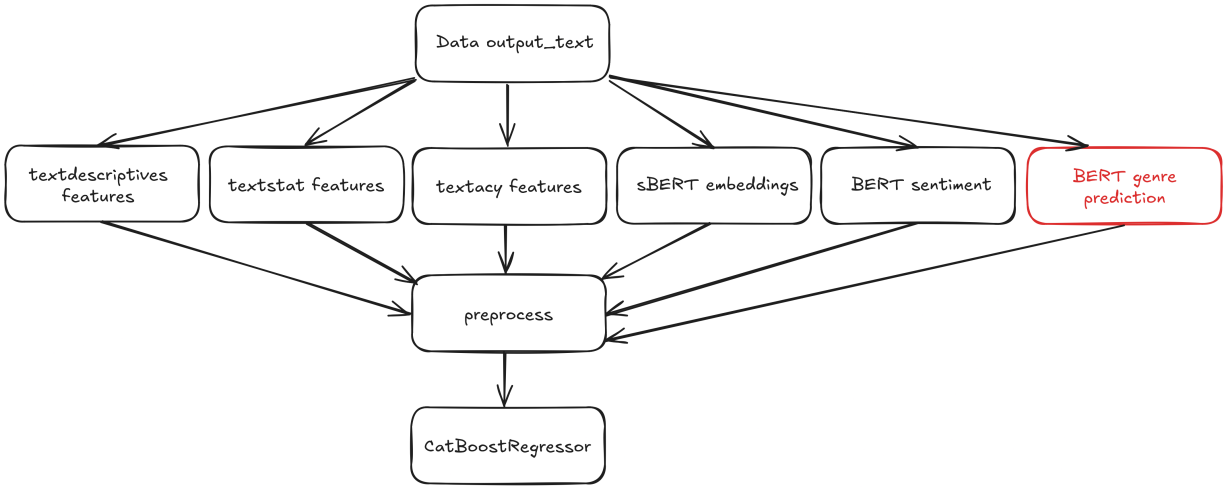
Future work

Что ещё пробовали:

- **Заместо CatBoost поверх фичей использовали simple NN**: Результаты показали, что CatBoost даёт лучшие результаты.
- **Заменили регрессию на классификацию**: Поделили диапазоны целевых значений (например, рейтинга) на бины и применили классификацию. Такой метод не улучшил результаты.
- **Поиск статей**: Провели быстрый поиск существующей литературы по данной теме, но не нашли каких-то нормальных готовых решений или метрик.

Что ещё можно сделать:

- Добавить различные профильные фичи, такие как типы рифм, виды четверостиший, тип стихотворной формы, количество рифм и другие элементы. Для этого потребуется знание поэзии и её теории.
- Сделать просто нормальный датасет с экспертными оценками стихов и зафайнтюнить LLM.
- Всё таки использовать жанр, так как для этого датасета он сильно влияет на качество. Но тогда нужно дополнительно обучить классификатор жанров, например BERT:



- Репозиторий с ноутбуками