**Министерство науки и высшего образования Российской Федерации**

**ФГАОУ ВО «УрФУ имени первого Президента России Б.Н. Ельцина»**

Кафедра «школа бакалавриата»

Оценка работы\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Руководитель от УрФУ Астафьева А.В.

Тема задания на практику

Оценка качества ответов больших языковых моделей

ОТЧЕТ

Вид практики Производственная практика

Тип практики Производственная практика, проектно-технологическая

Руководитель практики от предприятия (организации) Астафьева А.В.

**ФИО руководителя Подпись**

Студент Загидулин А.С.

**ФИО студента**

Специальность (направление подготовки) 10.03.01 Информационная безопасность

Группа РИ-311002

Оглавление

[Введение: 3](#_Toc173266508)

[Онтологии 3](#_Toc173266509)

[RAG 6](#_Toc173266510)

[Метрики оценки качества модели 7](#_Toc173266511)

[Исходные данные для оценки метрики: 7](#_Toc173266512)

[BLEU-метрика 8](#_Toc173266513)

[Rouge-метрика 9](#_Toc173266514)

[Семантическая похожесть 11](#_Toc173266515)

[BERTScore 13](#_Toc173266516)

[METEOR 14](#_Toc173266517)

[TER 15](#_Toc173266518)

[chrF, chrF++ 16](#_Toc173266519)

[Результат 17](#_Toc173266520)

# Введение:

NLP (Natural Language Processing) — это направление в машинном обучении, посвящённое распознаванию, генерации и обработке устной и письменной человеческой речи. Находится на стыке дисциплин искусственного интеллекта и лингвистики. Благодаря NLP компьютеры могут читать, интерпретировать и понимать человеческий язык, а также выдавать ответные результаты.

## Онтологии

Онтологии - формальное описание знаний из какого-то домена (предметной области) с учетом имеющихся сложных правил и связей между элементами, позволяющим сделать автоматическое извлечение знаний.

В математическом основании онтологии лежит так называемая дескриптивная логика (раздел математики), предполагающая, что любая информация, высказанная на естественном языке, может быть представлена в виде цепочки триплетов (Рисунок 1):

Изображение выглядит как круг, Шрифт, линия, диаграмма

Автоматически созданное описание

Рисунок 1 Цепочка триплетов

Отрывок из стихотворения «Дом, который построил Джек…» (пер. С.Я. Маршака).

*Вот****дом****,  
Который****построил Джек****.*

*А это****пшеница****,  
Которая в тёмном****чулане хранится*** *В****доме****,  
Который****построил Джек****.*

Описанные в стихах отношения между различными сущностями мы можем представить в виде онтологии (Рисунок 2).

Изображение выглядит как текст, снимок экрана, диаграмма, круг

Автоматически созданное описание

Рисунок 2 Пример онтологии

Онтология представляется в виде графа, вершины которого это сущности, а ребра – отношения между сущностями. Считается, что любое утверждение на естественном языке можно представить в виде простых предложений, из которых можно извлечь сущности и отношения между ними. Есть два основных инструмента: RDF (Resource Description Framework) или OWL (Ontology Web Language). OWL позволяет дополнительно описывать логические правила над данными. Онтологии (в отличие от обычных баз данных) позволяют находить скрытые данные. Обычные хорошо подходят для поиска конкретной информации, а базы знаний нужны там, где надо выявлять новые знания, например, в системах поддержки принятия решений (экспертных системах).

В этот момент ещё было непонятно, зачем именно оно используется, но разобрав пример стало ясно:

1. Доступ сотрудников в помещения

Рассмотрим ситуацию, в которой нам надо определить, в каком помещении находится каждый из сотрудников. При этом нам известно, что

* Иванов не имеет доступа в серверную и кабинет 101;
* Петров не имеет доступа в кабинет 101;
* Сидоров не имеет доступа в хранилище документов.

То есть у нас три сотрудника (Иванов, Петров, Сидоров) и три помещения (серверная, кабинет 101 и хранилище документов). Для однозначного решения задачи нам надо более строго определить свойства элементов онтологии, например, зафиксировать тот факт, что у нас нет других сотрудников и помещений и что каждый сотрудник имеет доступ ровно к одному помещению, в противном случае не удастся логически однозначно решить задачу. Принимая такие условия решение задачи становиться тривиальным и легко рассчитывается в уме: *Иванов-хранилище документов*, *Петров-серверная*, *Сидоров-кабинет 101*.

1. Задача Эйнштейна

Постановка задачи

1. На улице стоят пять домов.
2. Англичанин живёт в красном доме.
3. У испанца есть собака.
4. В зелёном доме пьют кофе.
5. Украинец пьёт чай.
6. Зелёный дом стоит сразу справа от белого дома.
7. Тот, кто курит Old Gold, разводит улиток.
8. В жёлтом доме курят Kool.
9. В центральном доме пьют молоко.
10. Норвежец живёт в первом доме.
11. Сосед того, кто курит Chesterfield, держит лису.
12. В доме по соседству с тем, в котором держат лошадь, курят Kool.
13. Тот, кто курит Lucky Strike, пьёт апельсиновый сок.
14. Японец курит Parliament.
15. Норвежец живёт рядом с синим домом.

Кто пьёт воду? Кто держит зебру?

В целях ясности следует добавить, что каждый из пяти домов окрашен в свой цвет, а их жители — разных национальностей, владеют разными животными, пьют разные напитки и курят разные марки американских сигарет. Ещё одно замечание: в утверждении 6 *справа* означает справа относительно вас.

Решение этой задачи уже нетривиальное и требует достаточно много времени на её решение “на листочке”. Но при помощи, например программы Protégé, при её правильной настройке решение находится довольно просто, Норвежец пьет воду, а Японец держит зебру.

## RAG

RAG (Retrieval-Augmented Generation "Поисковая расширенная генерация") - это техника, повышающая производительность языковых моделей путём предоставления модели контекста вместе с вопросом (Рисунок 3).

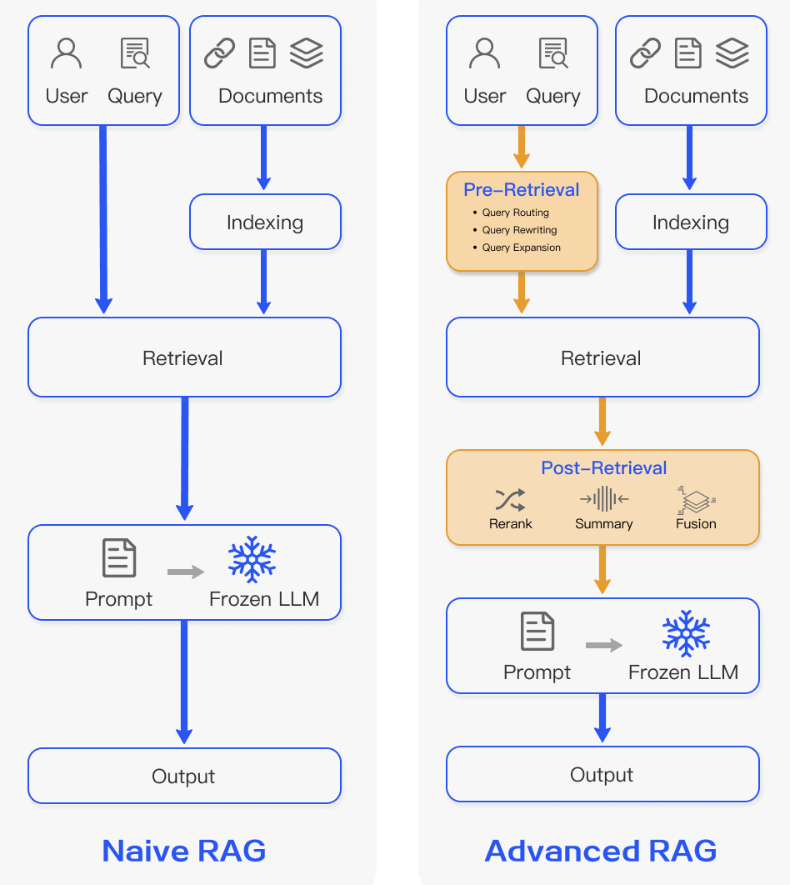


Рисунок 3 Схемы техники RAG

Один из способов повышения производительности это преиндексация (Pre-Indexing. Это предобработка текста, на котором обучается модель, например: оптимизация нарезки текста на чанки, приведение документов к единообразному виду, добавление метаданных.

Еще один способ это Post-Retrieval. Это обработка извлечённых сегментов, выделение каким-либо образом сегменты, которые более обносятся к заданной теме, к заданному вопросу. Т.е. на выходе не весь текст, а отдельно выделенные более важные его части, относящиеся к вопросу.

# Метрики оценки качества модели

## Исходные данные для оценки метрики:

Для оценки её качества были предоставлены вопросы с ответами специалистов на которые опирался при оценивании модели <https://187.ussc.ru/faq/>. Они были преобразованы в файл q\_data.xlsx, где вопросы находятся в столбце Question, а ответы в Answers. Отсюда вычленяются две переменные: для массива вопросов – questions, для ответов - answers. Ответы самой модели на вопросы questions находятся в переменной message. Код подготовки данных представлен на Рисунке 4.

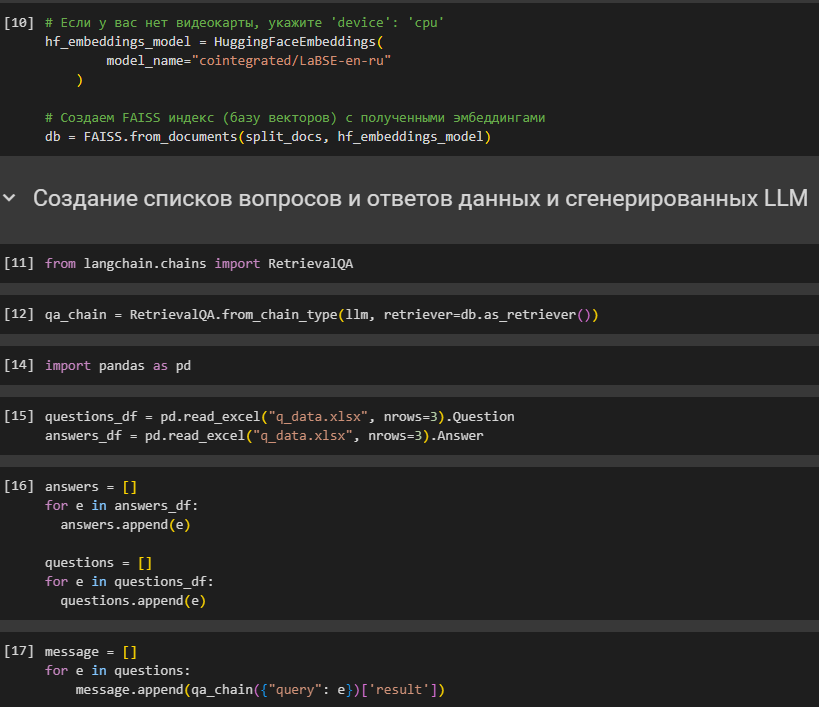


Рисунок 4 Подготовка данных

Из интересного можно заметить, что длины массивов questions[], answers[], message[] – одинаковые.

## BLEU-метрика

BLEU - метрика (Bilingual Evaluation Understudy) — это измерение различий между автоматическим переводом и эталонными переводами того же предложения, выполненными людьми. Эта метрика как раз подходит исходя из данных, которые мы получаем. Алгоритм BLEU сравнивает последовательные фразы автоматического перевода с последовательными фразами, которые он находит в эталонном переводе, в основном длина последовательных фраз не превышает четырёх слов, и взвешенно подсчитывает количество совпадений. Эти совпадения не зависят от позиции. Высшая степень совпадения указывает на более высокую степень сходства с эталонным переводом и более высокий балл. К итоговому значению применяется корректировка – штраф за краткость, чтобы избежать слишком высоких оценок BLEU для кратких и неполных переводов. Внятность и грамматика не учитываются.

Реализация метрики представлена на Рисунке 5.

Изображение выглядит как текст, снимок экрана, программное обеспечение, Мультимедийное программное обеспечение

Автоматически созданное описание

Рисунок 5 Реализация метрики BLEU

В процессе реализации была найдена проблема, что метрика выдавала значение результата равное 0, хоть на скриншоте она и очень близка к 0, но не о. А было это из-за того, что в метод corpus\_bleu() подавались массивы предложений, не поделённые на слова, и при тестировании массива из 3-х вопросов и ответов, предложения не были поделены, и либо у последовательных фраз максимальная длина была 1, либо сами члены массива становились последовательными фразами, связанными друг с другом, хотя это совсем не так.

## Rouge-метрика

ROUGE (Recall-Oriented Understudy for Gisting Evaluation) – это набор показателей и программный пакет, используемый для оценки задач суммаризации и машинного перевода в NLP. Метрика сравнивает созданные машиной ответы с ответом(-и), созданным человеком. Другими словами, как и BLEU, ROUGE основаны на подсчете совпадений слов и словосочетаний в машинном переводе и в эталоне. Но для ROUGE считается не только precision, но и recall, а также параметр F1. Это позволяет обойтись без штрафа за краткость. Разные метрики серии ROUGE различаются количеством слов, для которых ищутся совпадения:

ROUGE-N — это компонент оценки ROUGE, который количественно определяет перекрытие N-граммов, непрерывных последовательностей из N элементов (обычно слов или символов), между сгенерированной системой сводкой и справочной сводкой. Он дает представление о точности и повторяемости выходных данных системы, учитывая совпадающие N-граммовые последовательности.

ROUGE-L: вычисляет самую длинную общую подпоследовательность (longest common subsequence) между сгенерированным и эталонным текстами.

ROUGE-S фокусируется на пропуск-биграммах. Пропуск-биграмма - это пара слов в предложении, которая допускает пробелы или слова между ними. Этот компонент определяет перекрытие пропусков и биграмм между системным и справочным резюме, позволяя оценить сходство структуры на уровне предложений. Он может фиксировать перефразирующие связи между предложениями и давать представление о способности системы передавать информацию с гибким порядком слов.

Реализация метрики представлена на Рисунке 6.

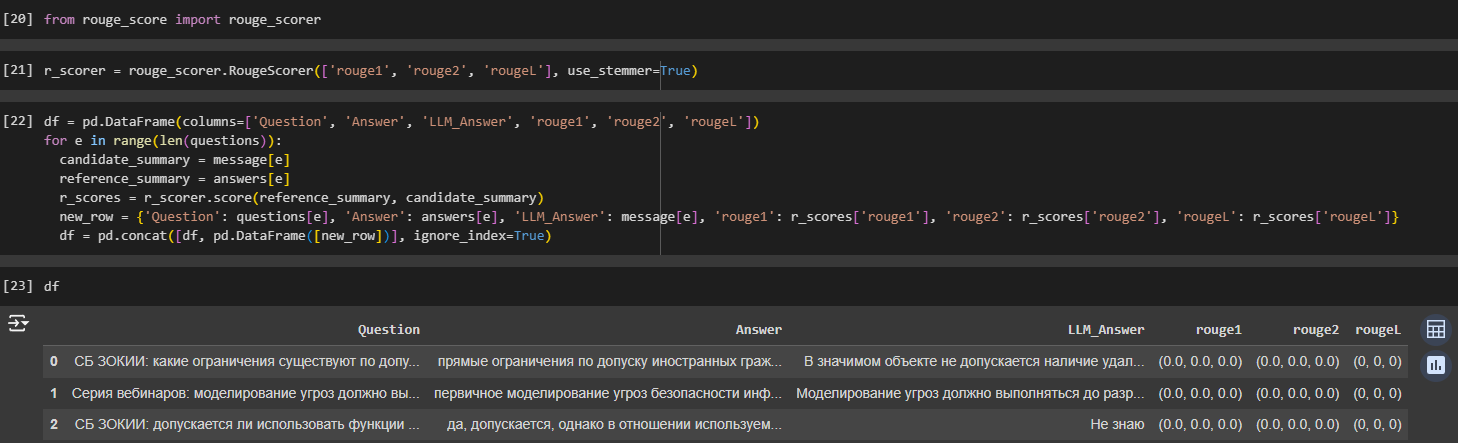


Рисунок 6 Реализация метрики Rouge

Ответы, которая выдавала модель message были очень короткие, в отличие от ответов экспертов answers, а на 3-й вопрос модель ответила ‘не знаю’, поэтому результат метрики для каждого вопроса – 0. Три значений, потому что rouge1, rouge2, rougeL каждое имеет значения precision – точность, recall - повторяемость и fmeasure – F1 (Рисунок 7).

Изображение выглядит как текст, Шрифт, снимок экрана

Автоматически созданное описание

Рисунок 7 Словарь метрики rouge

## Семантическая похожесть

Семантическое сходство текстов — это метрика, которая оценивает насколько похожи два текста с точки зрения смысла. Ответы модели и эталонные значения кодируются в эмбеддинги с помощью одной из предварительно обученных моделей, в нашем случае "deepvk/USER-bge-m3".

Затем оценивается косинусное сходство ответа и эталонного значения Косинусное сходсттво - это мера сходства между двумя ненулевыми векторами, определенными в пространстве внутреннего произведения. Косинусное сходство принадлежит интервалу [0,1] - два пропорциональных вектора имеют косинусное сходство, равное 1, а два ортогональных вектора имеют сходство, равное 0.

Реализация метрики представлена на Рисунке 8.



Рисунок 8 Реализация метрики семантической похожести

В данном случае представлено среднее значение семантической похожести, но оно высчитывается для каждого вопроса отдельно, и на что можно обратить внимание, что вопрос, на который модель ответила ‘не знаю’, имеет семантическую похожесть с ответом эксперта, равную 0.42731145. Также, стоит отметить, что метрика сравнивает эмбединги предложений, что лучше подходит для сравнения ответов двух разных моделей, а не ответы модели с ответами, данными человеком. Но для того, чтобы можно было сравнить модель по-разному эта метрика и была реализована. Также, для увеличения точности можно изменять AutoTokenizer и AutoModel. В процессе реализации кода было подтверждено, что модели для английского языка работают гораздо хуже, что не удивительно, ведь обрабатываются документы на русском.

## BERTScore

BERTScore — это метрика для оценки качества генерации текста, основанная на использовании предобученной модели BERT (Bidirectional Encoder Representations from Transformers). Она предназначена для оценки сходства между сгенерированным текстом и эталонным текстом с использованием векторных представлений слов, полученных с помощью BERT.

BERTScore вычисляется на основе косинусного расстояния между эмбеддингами слов в сгенерированном и эталонном текстах, что очень напоминает семантическую близость:

Плюсы:

* BERTScore использует предобученную модель BERT, которая обучена на больших объемах текстовых данных и демонстрирует высокую производительность в различных задачах обработки естественного языка.
* Она предоставляет более точную оценку качества генерации текста по сравнению с классическими метриками, такими как BLEU или ROUGE.
* BERTScore учитывает семантическое сходство между сгенерированным и эталонным текстами, что делает ее более информативной и полезной в реальных приложениях.

Минусы:

* BERTScore требует предварительно обученной модели BERT, что может потребовать больших вычислительных ресурсов для ее применения.
* Она может быть более медленной в вычислении по сравнению с классическими метриками, так как включает в себя вычисление векторных представлений слов с помощью BERT.
* BERTScore может быть менее интерпретируемой, чем некоторые другие метрики, так как она основана на сложных моделях глубокого обучения.

Реализация метрики представлена на Рисунке 9.

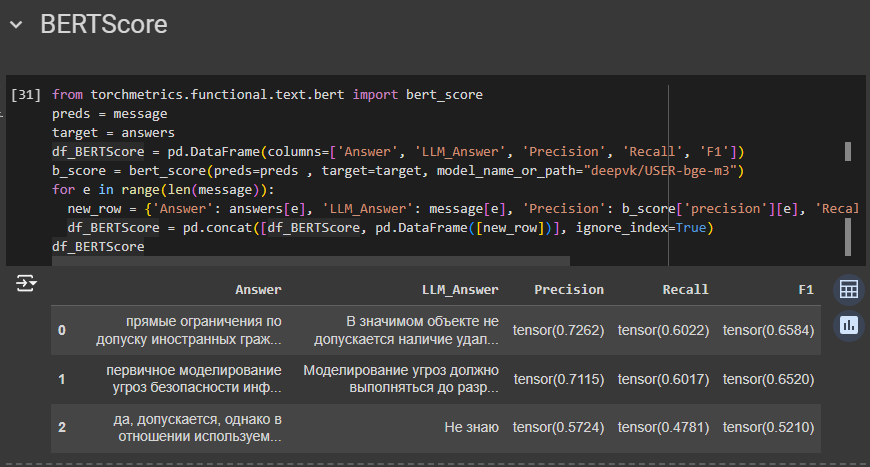


Рисунок 9 Реализация метрики BERTScore

Стоит отметить, что, на глаз, результаты метрики лучше, чем у семантической похожести. А также, как и у метрики ROUGE – BERTScore вычисляет Precision, Recall и F1.

## METEOR

METEOR (Metric for Evaluation of Translation with Explicit ORdering) – метрика качества машинного перевода, предложенная Banerjee and Lavie в 2005.

Оценка основана на подсчете совпадений отдельных слов (unigrams), c расчетом precision, recall, а также обобщающего показателя гармонического среднего (Fmean). В качестве совпадений учитываются не только точные соответствия слов (как в BLEU или ROUGE), но и наличие однокоренных слов или синонимов.

Итоговая метрика METEOR рассчитывается путем применения к Fmean штрафа за малую длину либо отсутствие совпадающих n-grams.

Могу предположить, что метрика основана на объединении метрик BLEU и семантической похожести, т.е. учитываются точные совпадения, но и в то же время учитывается сходство смысла слов.

Реализация метрики представлена на Рисунке 10.

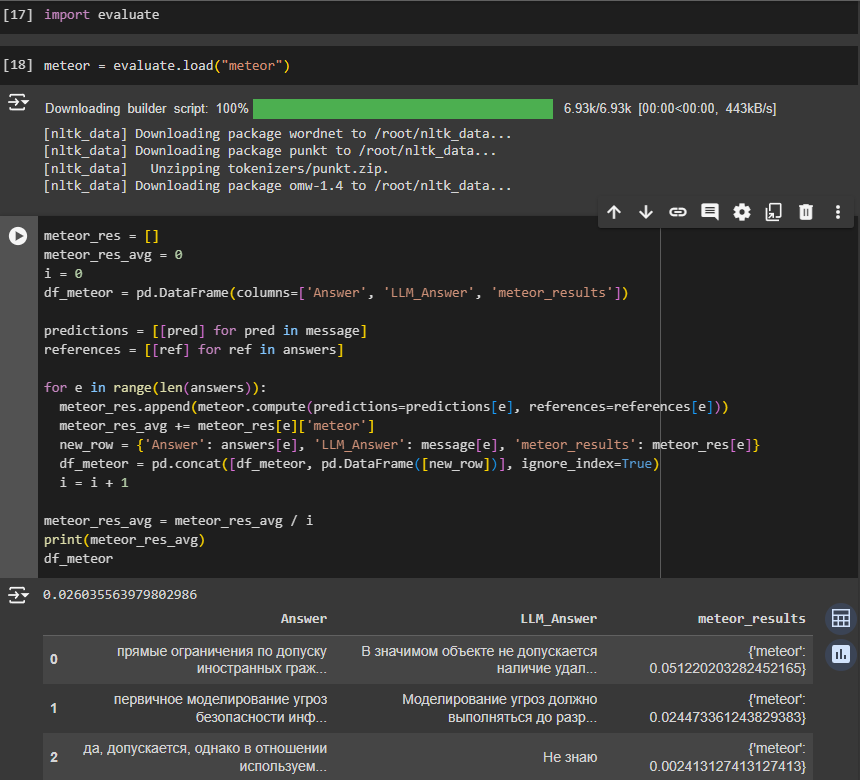


Рисунок 10 Реализация метрики METEOR

Благодаря тому, что метрика используем семантическую похожесть значение этой метрики получается гораздо выше, чем у метрики BLEU, хотя их смысл и похож, и различие как раз в семантической близости. Но если сравнить с семантической похожестью, то результаты будут гораздо хуже, ведь точных совпадений гораздо меньше. И на самом деле этот результат мне больше по душе, ведь семантическая похожесть ‘не знаю’ и ответа эксперта была примерно равна 0,4; а в метрике METEOR – 0,002.

## TER

TER (Translation Edit Rate) – метрика качества машинного перевода, предложенная Snover et al. в 2006. Основана на подсчете минимального числа правок, требуемых для приведения машинного перевода в полное соответствие наиболее близкому эталонному переводу. Подсчет числа правок автоматизирован.

Итоговый показатель TER определяется путем деления числа требуемых правок на усредненную по всем эталонам длину предложения:

Реализация метрики представлена на Рисунке 11.

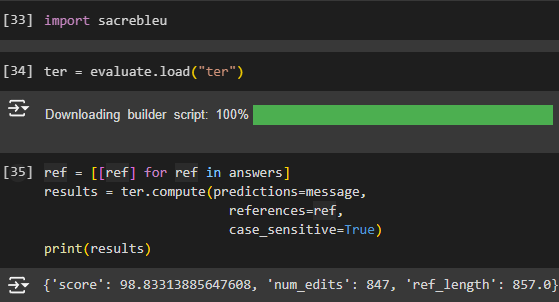


Рисунок 11 Реализация метрики TER

‘score’ – score TER = (num\_edits / sum\_ref\_lengths \* 100)

‘num\_edits’ - совокупное количество правок

‘ref\_length’ - совокупная средняя эталонная длина

Если score = 0, значит никаких правок не понадобилось. Чем выше значение score, тем хуже. Если score больше 100, то совокупное число правок num\_edits превышает совокупную длину ссылок ref\_length.

В данном случае оценка метрикой модели очень даже плохая, но опять же, для реализации метрики были взяты лишь первые 3 вопроса из одной из самых ранних моделей.

## chrF, chrF++

chrF (**c**orrelations with **h**uman **r**ankings for 6-gram **F**1-score) – метрика качества машинного перевода, предложенная Maja Popovic в 2015, затем была усовершенствована ею же до chrF++ в 2017.

chrF основана на подсчете совпадений последовательно идущих символов (character n-grams – в отличие от token n-grams для ROUGE) в машинном и эталонном переводах.

chrF++ учитывает как совпадения последовательно идущих символов, так и совпадения последовательно идущих слов.

В качестве итоговой метрики chrF и chrF++ выступает параметр F1.

Реализация метрики представлена на Рисунке 12.

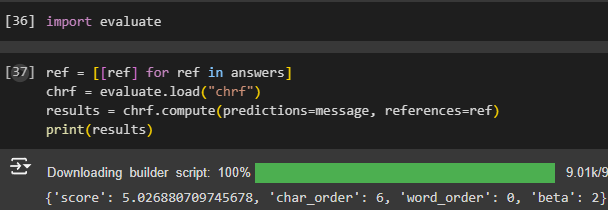


Рисунок 12 Реализация метрики chrF

‘score’ – score chrF (chrF++)

‘char\_order’ – порядок символов в n-граммах (по умолчанию установлено на 6)

‘word\_order’ – порядок слов в n-граммах (по умолчанию установлено на 0, если значение = 2, то метрика называется chrF++)

‘beta’ – определяет важность точности и повторяемовть (по умолчанию установлен на 2)

# Результат

По окончанию практики попросил у коллег модель, которая получилась у них по заданию и запустил метрики на ответах, полученных с помощью их модели. Результаты модели по разным метрикам представлены на рисунках 13-19.

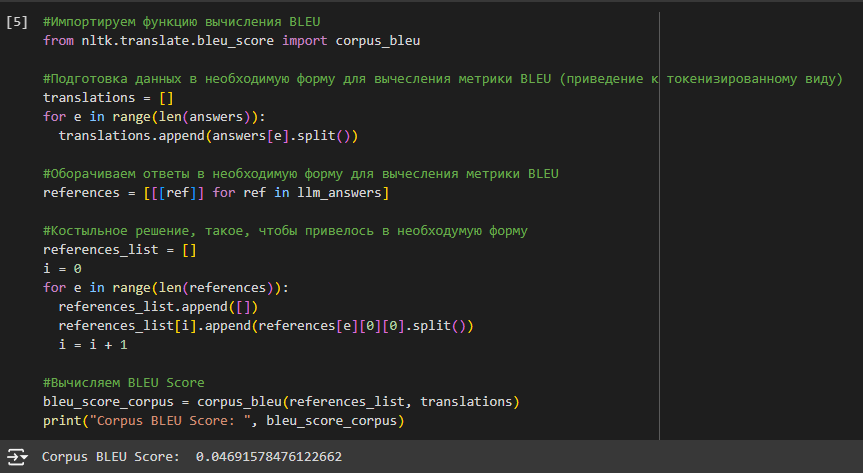


Рисунок 13 BLEU

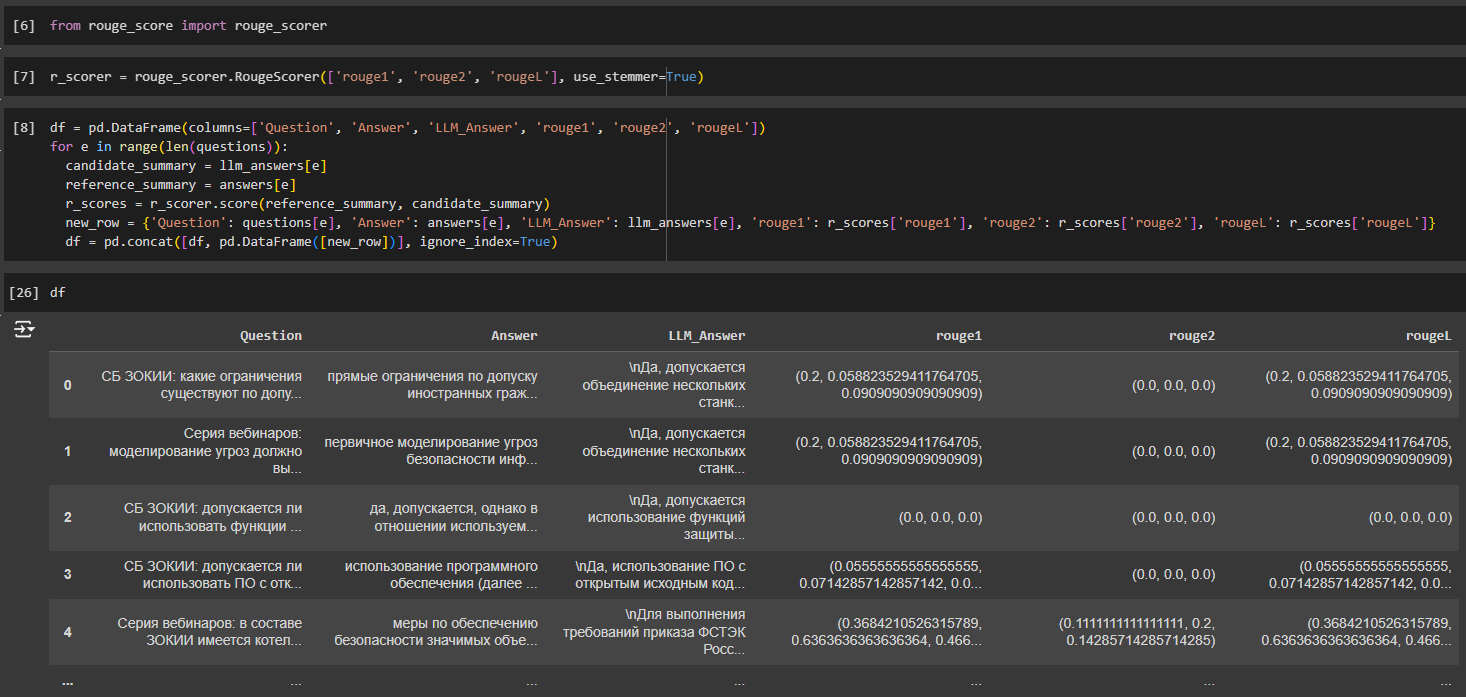


Рисунок 14 ROUGE

Изображение выглядит как текст, снимок экрана, программное обеспечение, Мультимедийное программное обеспечение

Автоматически созданное описание

Рисунок 15 Семантическая похожесть

Изображение выглядит как текст, снимок экрана, программное обеспечение, Мультимедийное программное обеспечение

Автоматически созданное описание

Рисунок 16 BERTScore

Изображение выглядит как текст, снимок экрана, программное обеспечение

Автоматически созданное описание

Рисунок 17 METEOR

Изображение выглядит как текст, снимок экрана, программное обеспечение, Мультимедийное программное обеспечение

Автоматически созданное описание

Рисунок 18 TER

Изображение выглядит как текст, Шрифт, программное обеспечение, Мультимедийное программное обеспечение

Автоматически созданное описание

Рисунок 19 chrF

Все метрики, кроме семантической похожести и TER выдают лучшие результаты, чем при тестировании на 3-х вопросах. Особенно радует, что улучшились результаты BLEU, ROUGE и METEOR, так как они специализируются на сравнении ответов, сгенерированных llm и ответов людей экспертов. Однако огорчает, что у метрики TER результат выше 100, ведь это значит, что требуется большое количество правок в ответы llm.

Попробовал изначальную модель, результаты представлены на рисунках 20-26.

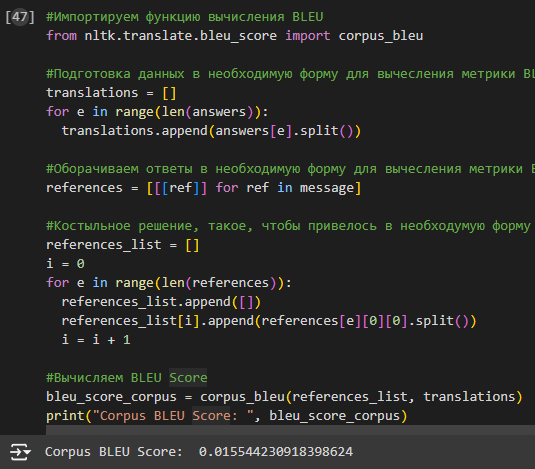


Рисунок 20 BLEU

Изображение выглядит как текст, снимок экрана, программное обеспечение, Мультимедийное программное обеспечение

Автоматически созданное описание

Рисунок 21 ROUGE

Изображение выглядит как текст, снимок экрана, программное обеспечение

Автоматически созданное описание

Рисунок 22 Семантическая похожесть

Изображение выглядит как текст, снимок экрана, программное обеспечение

Автоматически созданное описание

Рисунок 23 BERTScore

Изображение выглядит как текст, снимок экрана, программное обеспечение, Мультимедийное программное обеспечение

Автоматически созданное описание

Рисунок 24 METEOR

Изображение выглядит как текст, снимок экрана, Шрифт, программное обеспечение

Автоматически созданное описание

Рисунок 25 TER

Изображение выглядит как текст, Шрифт, снимок экрана

Автоматически созданное описание

Рисунок 26 chrF

Результаты, как и ожидалось хуже, чем на модели, полученной в конце практики у всех метрик, кроме метрики TER. Но по моему мнению это получилось, потому что у изначальной модели было много ответов ‘Не знаю’ (Рисунок 27), и их нужно меньше править.

Изображение выглядит как снимок экрана, Шрифт, текст, дизайн

Автоматически созданное описание

Рисунок 27 Количество ответов 'Не знаю'