**Как был проведен лингвистический анализ текста? и как получена оценка в 79.5 баллов?**

**Анализ привели в коде.**

**Анализ датасета был сделан по 6 ключевым параметрам:**

**1. Полнота данных: 100.00%**

**2. Среднее лексическое разнообразие: 0.843**

**3. Средний уровень читаемости: 61.50**

**4. Средняя структурная целостность: 0.963**

**5. Средний уровень отсутствия шума: 0.964**

**6. Средняя лингвистическая корректность: 0.500**

**Итоговая оценка качества данных была выведена из нормализованных метрик.**

**Нормализованные показатели (0-100):**

**Полнота данных: 100.0**

**Лексическое разнообразие: 84.3**

**Уровень читаемости: 61.5**

**Структурная целостность: 96.3**

**Отсутствие шума: 96.4**

**Лингвистическая корректность: 50.0**

**ОБЩАЯ ОЦЕНКА КАЧЕСТВА ДАННЫХ - 78.9 из 100 баллов.**

**Почему решили обрезать до топ-50 слов предложения?**

Решили обрезать до max\_len=50, потому что в основном это комментарии, и их длина не превышает 50 символов. Кроме того, для стандартизации: единый размер для batch-обработки. Также в комментариях основная смысловая нагрузка, как правило, сосредоточена в начале, а длинные тексты часто содержат повторы и "воду". Поэтому мы добились ускорения обучения, снижения потребления памяти без критической потери информации

**Интересен выбор num\_filters=100 в TextCNN\_FT - почему?**

Во-первых, ориентировались на Ваш файл, в котором было указано такое количество фильтров. Решили начать с него. Для нашего кейса значение этого гиперпараметра также подошло, его уменьшение приводило к ухудшению метрик, а увеличение — к улучшению метрик значимо не приводило. Во-вторых, это адекватное количество фильтров для 30к примеров, пропорционально распространенное.

**train\_model - перепишите через trainer, evaluate - тоже перепишите через trainer**

В Вашем задании было указано «классифицируйте текстовые данные одним из методов, которые вы рассматривали на вебинаре 4 «классификация текста»». Мы во всем ориентировались на подготовленный Вами файл к данному вебинару. Trainer к моменту данного вебинара мы еще не рассматривали, поэтому мы его не использовали.

**замерьте еще f1-score, так как у вас несбалансированный класс (Macro-F1)**

f1 ожидаемо хуже (к сожалению, полностью перезапустить код не получилось из-за ограничения по ресурсам, код тяжелый из-за фасттекста). Это решается формированием синтетических примеров на малопредставленный класс (можно в т.ч. через нейросетевые модели нагенерить), однако датасет слишком большой, чтобы мы могли в адекватные сроки нагенерировать такое количество примеров. Это идеально решалось бы локальной моделькой, но ни у кого из команды не тянет железо.

**не до конца понял идею того, что для русского датасета вы делите train-test пополам, а для английского 80-20**

Особой идеи в этом нет, просто увеличение обучающей выборки в русском датасете до стандартных 80% не привело к улучшению метрик, 50% было достаточно для выхода на плато, в то же время, увеличение тестовой выборки, исходя из центральной предельной теоремы, всегда делает выборку более репрезентативной и близкой к истинному распределению.

**Вопросы по 95% вариант:**

**Добавлены комментарии в файл с русским датасетом**