**> Введение**

В предложенном задании необходимо реализовать несколько метрик, которые были рассмотрены в лекции. Здесь встретятся как простые метрики, применяемые и вне LTR (для практики и закрепления материала), так и специфичные для ранжирования метрики, целью имплементации которых является проверка знаний и тонкостей "под капотом". Все функции, кроме вспомогательной compute\_gain, в качестве первых двух аргументов принимают на вход тензоры ys\_true и ys\_pred. Считайте, что это вещественные тензоры pytorch размерности n, задающие целевые отметки релевантности и предсказанные значения соответственно. Для генерации примеров входных данных можете использовать torch.rand(n), если не указана специфика исходных тензоров. В рамках задания подразумевается, что ys\_predсодержит в себе уникальные значения без повторений.  Для каждой метрики в шаблоне скрипта, прикреплённого к домашнему заданию, указан тип возвращаемого значения. Пожалуйста, обращайте на него внимание! Также удостоверьтесь, что значение метрики не выходит за рамки интервала (например, если метрика нормирована от 0 до 1).

Примечание: если вы не знакомы или малознакомы с PyTorch, то в общем случае можете обращаться с тензорами так же, как и с массивами numpy. Все необходимые импорты сделаны, дополнительные импорты делать запрещается. Из специфичных методов понадобится только сортировка — [ссылка](https://pytorch.org/docs/stable/generated/torch.sort.html?highlight=sort#torch.sort) на документацию.

**> Детали реализации и описание**

num\_swapped\_pairs — функция для расчёта количества неправильно упорядоченных пар (корректное упорядочивание — от наибольшего значения в ys\_true к наименьшему) или переставленных пар. Не забудьте, что одну и ту же пару не нужно учитывать дважды.

compute\_gain — вспомогательная функция для расчёта DCG и NDCG, рассчитывающая показатель Gain. Принимает на вход дополнительный аргумент — указание схемы начисления Gain (gain\_scheme).

В лекции был приведён пример константного начисления, равного в точности оценке релевантности. Необходимо реализовать как этот метод (при gain\_scheme="const") начисления Gain, так и экспоненциальный (gain\_scheme="exp2"), рассчитываемый по формуле (2r−1), где r — реальная релевантность документа некоторому запросу. Логика здесь такова, что чем выше релевантность, тем ценнее объект, и темп роста “ценности" нелинейный — гораздо важнее отделить документ с релевантностью 5 от документа с релевантностью 3, нежели 3 от 1 (ведь они оба слабо релевантны).

Функция принимает на вход единственное число (не тензор).

dcg и ndcg — функции расчёта DCG и NDCG соответственно. Принимают на вход дополнительный параметр gain\_scheme, аналогичный таковому в функции compute\_gain (её необходимо использовать при расчётах этих метрик). Для NDCG разрешается переиспользовать функцию расчёта DCG.

precission\_at\_k — функция расчёта точности в топ-k позиций для бинарной разметки (в ys\_true содержатся только нули и единицы). Если среди лейблов нет ни одного релевантного документа (единицы), то необходимо вернуть -1. Функция принимает на вход параметр k, указывающий на то, по какому количеству объектов необходимо произвести расчёт метрики. Учтите, что k может быть больше количества элементов во входных тензорах. При реализации precission\_at\_k необходимо добиться того, что максимум функции в единице был достижим при любом ys\_true, за исключением не содержащего единиц (попробуйте рассмотреть проблему на примере ранжирования поисковой выдачи, где в разметке присутствует n<k релевантных документов). В силу этой особенности не рекомендуется переиспользовать имплементацию precission\_at\_k в других метриках настоящего ДЗ. Можете также считать, что в ys\_true содержатся все возможные единицы в датасете (например, если система отбора кандидатов для ранжирования их пропустила, то можно добавить такой объект c предсказанным значением −∞).

reciprocal\_rank — функция для расчёта MRR (без усреднения, т.е. для одного запроса и множества документов). В ys\_true могут содержаться только нули и максимум одна единица.

p\_found — функция расчёта P-found от Яндекса, принимающая на вход дополнительный параметр p\_break — вероятность прекращения просмотра списка документов в выдаче. Базовая вероятность просмотреть первый документ в выдаче (pLook[0] в лекции) равняется единице. ys\_true нормированы от 0 до 1 (вероятность удовлетворения запроса пользователя).

average\_precision — функция расчёта AP для бинарной разметки (в ys\_true содержатся только нули и единицы). Если среди лейблов нет ни одного релевантного документа (единицы), то необходимо вернуть -1.