**Кейс “Предсказание рейтинга лекарства по обзорам”**

|  |  |
| --- | --- |
| **Тьютор команды** | Руднев Никита Владимирович |
| **Название кейсового задания** | Модель предсказания рейтинга лекарства по обзорам |
| **Краткое описание кейсового задания** | Создание модели для предсказания рейтинга по обзорам, вопросам и иным данным, предоставляемым пользователями сайта. |
| **Полное описание кейсового задания** | На основе данных, содержащих отзывы, скачанные с сайта с обзорами на лекарства разработать модель, позволяющую прогнозировать рейтинг лекарств.  Необходимо выделить у всеразличных лекарств их рейтинги по десятибалльной шкале, наиболее рекомендованные лекарства от той или иной болезни будут рассматриваться как лучшие среди своего сегмента.  Входные данные представляют собой наборы признаков по обзорам на лекарства разных пользователей.  Результатом предсказания модели должен стать рейтинг (от 1 до 10.) по каждому лекарству – оценка пользователей качества лекарства.  В зависимости от способа решения задачи результат может оцениваться различными метриками (можете выбрать их сами, в основном бог любит троицу) |
| **Проблематика** | После рекомендации врача, одной из главных ролей в покупке лекарств является общий рейтинг. На каждое лекарство существует несколько аналогов, дороже – дешевле, действенней – слабее. Как и в покупке товаров временами очень сложно найти самое лучшее для себя решение, ведь не редко доктора рекомендуют дорогие лекарства, которые могут быть и не по карману, а отзывы могут не сопровождаться оценкой, или в отдельном случае: модераторы отзывов могут оставить текст отзыва, но изменить оценку для повышения общей статистики товаров.  Именно для такого рода случаев мы создаём модель, которая будет определять рейтинг лекарства по отзыву чтобы максимизировать честность оценок и избежать разногласий между рейтингом и текстовым обзором.  Это позволит нам лучше оценивать тот или иной товар на основе результатов обученной модели. |
| **Образ результата** | В итоге разработанная модель машинного или глубокого обучения, позволяющая на основе имеющихся данных об обзорах на препараты рассчитать предположительный рейтинг лекарства.  В качестве подтверждения полученного результата ожидается ipynb и csv-файлы форматов **Report.ipynb** и **result.csv** соответственно. В файле Report.ipynb необходимо составить отчёт о проделанной работе над данными (предобработка, анализ данных, обучение модели и подбор параметров). Result.csv –колонка предсказанных моделью переменных rating csv файла test.csv (вы предсказываете колонку rating на тестовых данных и сохраняете эти предсказанные значения колонки в csv файл Result.csv).  Подготовленная презентация разработанной модели, содержащая основные аспекты:   * подход к решению задачи * подробно изложенный процесс моделирования * полученные результаты * пути к улучшению решения и/или интерпретация задачи, свои идеи по поводу задач в сфере лекарств |
| **Стек технологий, рекомендуемых к использованию** | Jupyter notebook, Python, scikit-learn, pandas, Data Science, Машинное обучение |
| **Ссылка на датасет** | <https://disk.yandex.ru/d/-0w_EXpRbnrRGQ>  **train.csv** - sample данных обучающего датасета.  **test.csv** - sample данных тестового датасета.  Имеются признаки:   * uniqueID - уникальный индикатор обзора * drugName - название лекарства * condition - состояние пациента * review - обзор в виде текста * rating - рейтинг (число от 1 до 10) * date - дата публикации обзора * usefulCount - полезность обзора (сколько лайков поставили другие пользователи обзору на сайте) |
| **Мощности** | Google Colab (бесплатный доступ) |
| **Необходимые данные, дополнения/пояснения/уточнения** ***Информация для участников хакатона, которая поможет разобраться в проблематике и предложить качественные решения кейсового задания*** | В работе приветствуется полнота, анализ данных и некоторые сделанные выводы о них, применение большего количества различных способов решения задачи.  При оценке будут учитываться точность модели, использованные подходы к предобработке данных, отбора признаков, выбранный алгоритм построения модели.  Примерные вопросы анализа данных в нашей задаче, решающиеся и предоставляемые в виде визуализаций или группировок:  Какие лекарства представлены? На какие болезни они направлены? Какие лекарства чаще всего используются для наиболее частых болезней? Как зависит качество лекарства в обзорах людей от его популярности? Какие еще можно сделать выводы из группировки?  Подсказки в выполнении:  Представление текста в виде матриц по типу tf-idf использует разреженные матрицы, не пытайтесь преобразовать их в numpy array методом .toarray() - скорее всего вылетит либо питон с memory error или среда или ваш компьютер!  Обычное PCA из sklearn не работает с разреженными матрицами scipy. Используйте TruncatedSVD, синтаксис идентичен, логика работы почти та же, но поддерживает sparced.  Оптимальный порог кумулятивной суммы описанной компонентами (суммарная доля описанной дисперсии) можно понизить до 0.8 . Приблизительно 4000 компонент должно хватить, однако вычисления могут занять какое-то время (минут 5-10). По умолчанию преобразователи #scikit-learn возвращают массив NumPy. Это может быть проблемой, если для последующих этапов обработки данных вам нужен датафрейм pandas.  К счастью, начиная с scikit-learn версии 1.3.2, вы можете использовать метод set\_output для получения результатов в формате датафреймов pandas.    Обучение алгоритмов может занять значительное время, даже для 4000 элем. Укажите, для начала, небольшое количество итераций град спуска (параметр max\_iter), например 100, чтоб понимать сколько времени он обучается у вас.  Можно пробовать параллельные алгоритмы сторонними библиотеками, например, dask.  Будет плюсом, если презентация будет отражать следующую информацию:   * Какой предварительный анализ данных проводился? Есть ли какие-то интересные выводы? * Какие подходы к отбору признаков пробовали? Какой в итоге выбрали и почему? * Добавляли ли новые фичи? * Какую валидационную схему использовали? * Как подбирали гиперпараметры? Какие параметры сильно влияли на метрики? * Какие алгоритмы машинного обучения пробовали и какой выбрали для итоговой модели? * Работает ли модель стабильно? Наблюдалось ли переобучение? Как с ним боролись? * С какими сложностями столкнулись? |