Классификация текстов новостей: от сбора данных до обучения модели

Легенда:
Вася увлекается хоккеем, а его брат Ваня — киберспортом. Вместе они решили создать систему, которая собирает новости **на русском языке** по их интересам и классифицирует их на три категории: **спорт**, **киберспорт** и **прочие тексты**. У них есть общий план, но они столкнулись с несколькими сложными задачами. Помогите братьям разработать систему, следуя шагам.

Время выполнения всех задач: 4 часа

Задача 1. Сбор данных

Легенда:
Вася предложил брать новости из групп по хоккею и спортивных пабликов, а Ваня — из сообществ по CS:GO и Dota 2. "А давай напишем программу, которая сама будет собирать новости из разных мест!" — предложил Ваня. "И обязательно нужно добавить обычные новости для сравнения", — добавил Вася.

Требования к результатам:

1. CSV-файл collected_data.csv со следующими колонками:
    - **id**: уникальный идентификатор текста
    - **text**: исходный текст
    - **category**: метка категории (спорт/киберспорт/прочее)
    - **source**: источник текста
    - **length**: длина текста
    - **date**: дата публикации (YYYY-MM-DD)

Пример структуры файла collected_data.csv:

```
id,text,category,source,length,date
1,"Сборная России по хоккею одержала победу над командой Финляндии со счётом 3:2 в матче Евротура. Решающую шайбу забросил Капризов на последней минуте встречи.",спорт,https://t.me/sportnews,156,2023-12-15
2,"Team Spirit разгромила Na'Vi со счётом 2:0! 🏆 #Dota2",киберспорт,https://t.me/cybersport,96,2024-01-20
3,"В Москве открылась новая станция метро. На церемонии открытия присутствовал мэр города. Станция будет обслуживать около 85 000 пассажиров в день.",прочее,https://t.me/news,135,2023-08-10
```

Критерии оценки (100 баллов):

1. **Объем и баланс данных (40 баллов):**
    - 500 ± 5% текстов для каждой категории (30 баллов)
    - Равномерное распределение по датам 2023-2024 гг. (10 баллов)
2. **Разнообразие источников (30 баллов):**
    - Минимум 3 источника для каждой категории
    - Не более 30% текстов из одного источника
3. **Качество текстов (25 баллов):**

- Длина текстов 200-3000 символов
- Отсутствие дубликатов
- Корректность дат и форматов

4. **Отчет (5 баллов):**
    - Файл report_task1.txt с описанием:
        - Методы сбора данных
        - Листинг основного кода

Задача 2. Обработка текстов

Легенда:
Когда братья посмотрели на собранные новости, они обнаружили много лишнего: эмодзи, ссылки, теги. "Надо всё это почистить, но сохранить названия команд и турниров", — решили они.

Требования к результатам:

1. CSV-файл processed_data.csv со следующими колонками:
    - **id**: идентификатор исходного текста из collected_data.csv
    - **cleaned_text**: текст после базовой очистки
    - **lemmatized_text**: текст после лемматизации
    - **entities**: сохраненные именованные сущности (JSON-строка)

Пример структуры файла processed_data.csv:

```
id,cleaned_text,lemmatized_text,entities
2"Team Spirit разгромила NaVi со счетом 2:0","team spirit разгромить navi со счет","[{""type"": ""TEAM"", ""text"": ""Team Spirit""}, {""type"": ""TEAM"", ""text"": ""NaVi""}]"
10,"Овечкин забросил 822-ю шайбу в карьере НХЛ. Великий россиянин продолжает погоню за рекордом Гретцки","овечкин забросить шайба в карьера нхл великий россиянин продолжать погоня за рекорд гретцки","[{""type"": ""PERSON"", ""text"": ""Овечкин""}, {""type"": ""ORG"", ""text"": ""НХЛ""}, {""type"": ""PERSON"", ""text"": ""Гретцки""}]"
```

Критерии оценки (100 баллов):

1. **Базовая очистка (35 баллов):**
    - Удаление HTML и спецсимволов (15 баллов)
    - Корректная работа с пробелами и пунктуацией (10 баллов)
    - Сохранение значимых именованных сущностей (20 баллов)
2. **Качество лемматизации (30 баллов):**
    - Корректность словарных форм (30 баллов)
3. **Техническое качество (20 баллов):**

- Корректность форматов (10 баллов)
- Сохранение всех исходных текстов (10 баллов)

4. **Отчет (5 баллов):**
    - Файл report_task2.txt с описанием:
        - Использованные методы очистки
        - Выбранные библиотеки лемматизации
        - Листинг основного кода

Задача 3. Обучение классификатора и классификация тестовой выборки

Легенда:
"Как научить компьютер отличать спортивные новости от киберспортивных?" — задумались братья. Они решили попробовать разные подходы, чтобы выбрать лучший. После обучения модели нужно проверить, насколько хорошо она работает на новых текстах.

Задание состоит из двух частей:

1. Обучение классификатора на предоставленных данных
2. Классификация текстов из тестового набора

Классы для классификации:

- 0: прочие тексты
- 1: спортивные новости
- 2: новости киберспорта

Входные данные:

- Файл test_data.csv с колонками:
    - **id**: уникальный идентификатор текста
    - **text**: текст для классификации

Требования к результатам:

1. CSV-файл predictions.csv с предсказаниями для тестовой выборки:
    - **id**: идентификатор текста из тестового набора
    - **predicted_class**: предсказанный класс (0, 1 или 2)

Пример структуры файла predictions.csv:

```
id,predicted_class
0,1
1,2
2,0
3,1
4,2
```

2. JSON-файл model_report.json с информацией об обучении:

Пример структуры файла model_report.json:

```
{
  "model_description": {
    "vectorization": "TF-IDF",
    "classifier": "CatBoostClassifier",
    "preprocessing": [
      "removing_stopwords",
      "lemmatization",
      "lowercase_conversion"
    ]
  },
  "training_process": {
    "split_ratio": "80/20",
    "cross_validation_scores": [0.85, 0.83, 0.86, 0.84, 0.85]
  }
}
```

3. Два файла с результатами, predictions.csv и model_report.json, необходимо архивировать с использованием ZIP-архиватора в файл **solution.zip**, который затем требуется отправить.

Критерии оценки (100 баллов):

1. **Качество предсказаний (75 баллов)**
    - Тестовая выборка содержит 75 текстов
    - За каждый правильно классифицированный текст начисляется 1 балл
2. **Технические требования (20 баллов)**
    - Корректный формат файла predictions.csv (5 баллов)
    - Предсказания для всех текстов из тестовой выборки (5 баллов)
    - Корректный формат model_report.json (5 баллов)
    - Полнота описания модели в отчёте (5 баллов)
3. **Отчёт по обучению модели (5 баллов)**

- Файл report_task3.txt с описанием:
    - Обоснование выбора методов предобработки
    - Обоснование выбора модели
    - Листинг основного кода

Формат сдачи

Участники должны предоставить следующие файлы:

1. collected_data.csv
2. processed_data.csv
3. predictions.csv
4. model_report.json
5. report_task1.txt
6. report_task2.txt
7. report_task3.txt

Автоматическая проверка

- Проверка форматов файлов
- Подсчет количественных показателей
- Валидация структуры данных
- Проверка корректности дат и источников
- Расчет метрик качества моделей

Жюри оставляет за собой право снизить или аннулировать баллы в случае выявления следующих нарушений в ходе ручной проверки:

1. Использование технологий генеративного искусственного интеллекта.
2. Отсутствие реальных ссылок на сообщения.
3. Изменение даты публикации сообщения.
4. Несоблюдение условий задания, включая использование сообщений на языке, отличном от русского.
5. Иные нарушения, не указанные явно, но противоречащие установленным правилам и критериям оценки.