News Text Classification: From Data Collection to Model Training

Background: Vasya is passionate about hockey, and his brother Vanya is into esports. Together, they decided to create a system that collects **Russian-language** news related to their interests and classifies them into three categories: **sports**, **esports**, and **other texts**. They have a general plan but have encountered several challenges. Help the brothers develop the system by following the steps below.

Time to complete all tasks: 4 hours

Task 1. Data Collection

Background: Vasya suggested collecting news from hockey groups and sports communities, while Vanya proposed gathering data from CS:GO and Dota 2 communities. "Let's write a program that automatically collects news from different sources!" Vanya suggested. "And we should definitely add regular news for comparison," Vasya added.

Requirements for the results:

1. A CSV file named `collected_data.csv` with the following columns:
    - **id**: Unique text identifier
    - **text**: Original text
    - **category**: Category label (спорт/киберспорт/прочее)
    - **source**: Text source
    - **length**: Text length
    - **date**: Publication date (YYYY-MM-DD)

Example structure of `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
```

Evaluation criteria (100 points):

1. **Data volume and balance (40 points):**

    - 500 ± 5% texts for each category (30 points)
    - Uniform distribution of dates across 2023-2024 (10 points)
2. **Diversity of sources (30 points):**
    - At least 3 sources for each category
    - No more than 30% of texts from a single source
3. **Text quality (25 points):**

- Text length between 200-3000 characters
- No duplicates
- Correct dates and formats

4. **Report (5 points):**
    - A file named `report_task1.txt` containing:
        - Data collection methods
        - Main code listing

Task 2. Text Processing

Background: When the brothers looked at the collected news, they noticed a lot of unnecessary elements: emojis, links, and tags. "We need to clean this up but preserve team and tournament names," they decided.

Requirements for the results:

1. A CSV file named `processed_data.csv` with the following columns:
    - **id**: Identifier from `collected_data.csv`
    - **cleaned_text**: Text after basic cleaning
    - **lemmatized_text**: Text after lemmatization
    - **entities**: Preserved named entities (JSON string)

Example structure of `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""}]"
```

Evaluation criteria (100 points):

1. **Basic cleaning (35 points):**
    - Removal of HTML and special characters (15 points)
    - Correct handling of spaces and punctuation (10 points)
    - Preservation of significant named entities (20 points)

2. **Lemmatization quality (30 points):**
    - Correct dictionary forms (30 points)
3. **Technical quality (20 points):**

- Correct formats (10 points)
- Preservation of all original texts (10 points)

4. **Report (5 points):**
    - A file named `report_task2.txt` containing:
        - Cleaning methods used
        - Lemmatization libraries chosen
        - Main code listing

Task 3. Training a Classifier and Classifying Test Data

Background: "How can we teach a computer to distinguish sports news from esports news?" the brothers wondered. They decided to try different approaches to choose the best one. After training the model, they need to test how well it performs on new texts.

The task consists of two parts:

1. Training a classifier on the provided data
2. Classifying texts from the test dataset

Classes for classification:

- 0: Other texts
- 1: Sports news
- 2: Esports news

Input data:

- A file named `test_data.csv` with columns:
    - **id**: Unique text identifier
    - **text**: Text to classify

Requirements for the results:

1. A CSV file named `predictions.csv` with predictions for the test dataset:
    - **id**: Identifier from the test dataset
    - **predicted_class**: Predicted class (0, 1, or 2)

Example structure of `predictions.csv`:

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

```
4,2
```

2. A JSON file named `model_report.json` with information about the training process:

Example structure of `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. The two files, `predictions.csv` and `model_report.json`, must be zipped into a file named **solution.zip**, which should then be submitted.

Evaluation criteria (100 points):

1. **Prediction quality (75 points):**
    - The test dataset contains 75 texts
    - 1 point is awarded for each correctly classified text
2. **Technical requirements (20 points):**
    - Correct format of `predictions.csv` (5 points)
    - Predictions for all texts in the test dataset (5 points)
    - Correct format of `model_report.json` (5 points)
    - Completeness of the model description in the report (5 points)
3. **Training report (5 points):**

- A file named `report_task3.txt` containing:
    - Justification of preprocessing methods
    - Justification of the chosen model
    - Main code listing

Submission Format

Participants must submit the following files:

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`

Automatic Evaluation

- File format validation
- Calculation of quantitative metrics
- Data structure validation
- Validation of dates and sources
- Calculation of model quality metrics

The jury reserves the right to deduct or nullify points in case of the following violations during manual review:

1. Use of generative AI technologies.
2. Absence of real message links.
3. Alteration of publication dates.
4. Non-compliance with task requirements, including the use of non-Russian language texts.
5. Other violations not explicitly stated but contrary to the established rules and evaluation criteria.