Instructions for Kaggle Competition 2024

IFT6390B

October 14, 2024

1 Description

In this project, you will take part in a Kaggle competition for text classification. The goal is to design a machine learning algorithm that can automatically sort short text documents into a pre-determined set of categories. In order to maintain the anonymity of the dataset and ensure a fair competition, we do not disclose the name of the dataset and do not provide original document entries. Instead, you are given document-wise word (term) count vectors as features, where each index is the number of times that a given term is present in the document. Alongside these vectors, you are also given a vocabulary map that corresponds to each index to a term. Your goal is to leverage this term count matrix to solve a binary text classification task. The evaluation metric is the macro F1 score on the test set. In summary, you are given the following data:

- `data_train.npy` - This is a NumPy array representing term vector counts for training, where each row corresponds to a document and each column represents a term in the vocabulary. The values (mostly 0s) indicate the count of each term in the respective document, forming a sparse matrix.
- `data_test.npy` - A similar NumPy array for testing. You need to create labels for this test set and submit it for evaluation.
- `vocab_map.npy` - Contains a mapping between the terms (words) and their corresponding indices in the term vector matrix.
- `labels_train.npy` - Contains the labels or target values for the training dataset (0 or 1).

2 Participation

For the graduate section (IFT6390B), the task must be solved individually and without the help of other students. In order to participate in the competition, you should:

- Create a Kaggle account if you do not have one already.
- Enter the competition using the following invitation link: `https://www.kaggle.com/t/b156d192b9374a549cc887d465431873`.
- From now on, you can access the competition via `https://www.kaggle.com/competitions/classer-le-text/`.

Important note: The maximum number of submissions is two per day. Therefore, you will not be able to spam submissions.

3 First milestone: Beat the baseline (Oct 21st)

You can see two baseline scores on the leaderboard. The first score corresponds to a classifier that assigns random labels to each document. The second baseline corresponds to a vanilla logistic regression classifier. For the first milestone, you will need to beat the baseline logistic regression classifier on the public leaderboard.

Important note: To beat the baseline, you are NOT allowed to use any machine learning library, e.g. `scikit-learn`. You should implement your solution from scratch using only NumPy and basic Python functionalities.

4 Second milestone: Compete (Nov 9th)

You have until **Nov 9th 23:59** to achieve the best performance you can on the task. In this phase you are free to implement any method you think would work best. The Kaggle leaderboard has a public and private component to prevent participants from "overfitting" to the leaderboard. The public leaderboard shows your score calculated on 30% percent of the test set, while the private leaderboard is based on your score on the 70% remainder of the test set. You are only able to see the public leaderboard during the competition. The points for this phase will be given based on your ranking on the private leaderboard that will be released at the end of the competition.

Important note: You must submit two separate solutions, one for the first phase (beating the baseline), and one for the second phase (your best-performing model). You should name your submission files to distinguish between the two. For your code submission on Gradescope, you should also separate the two solutions.

5 Third milestone: Submit Code and Report (Nov 12th)

You must write up a report that details your machine learning pipeline, including preprocessing, algorithms, optimization and learning, hyperparameter tuning, and validation procedure. You should also provide and compare the results of other methods you implemented before reaching the best-performing model. The report should contain the following elements. You will lose points if you do not follow these guidelines.

- Project title
- Your team name on Kaggle, as well as the list of team members, including their full name and student number (for graduate students each team has only one member).
- Introduction: briefly describe the problem and summarize your approach and results.
- Feature Design: Describe and justify your pre-processing and feature extraction methods.
- Algorithms: Give an overview of the learning algorithms used without going into too much detail.
- Methodology: Include any decisions about training/validation split, regularization strategy, optimization tricks, setting hyperparameters, etc.
- Results: Present a detailed analysis of your results, including graphs and tables where appropriate. This analysis should be broader than just the Kaggle results: include a short comparison of different values for important hyperparameters in your best-performing algorithm and also compare the performance of this method with at least two other methods you implemented.
- Discussion: Discuss the pros/cons of your approach and suggest ideas for improvement.
- References (very important if you use ideas and methods that you found in some paper or online; it is a matter of academic integrity).
- Appendix (optional). Here you can include additional results, more details of the methods, etc.

The main text of the report should not exceed 6 pages. References and appendix can be in excess of the 6 pages.

Your must submit your code (first and second milestones) and report (third milestone) on Gradescope before **Nov 12th, at 23:59**.

Submission Instructions

- You must have separate .py files/notebooks for the first and second milestones. The code must be well-documented. If you are not using Jupyter notebooks, you should include a README file containing instructions on how to run the code. You will need to submit a zip file containing your code and related files to Gradescope.
- The prediction file containing your predictions on the test set should only be submitted to Kaggle.
- The report in pdf format (written according to the general layout described earlier) should be submitted to Gradescope.

6 Evaluation Criteria

1. You will receive a minimum number of points if you beat the logistic regression baseline on Kaggle's public leaderboard (conditioned on your adherence to the aforementioned instructions).

2. You will be graded depending on the quality and technical soundness of your final report.

3. You will receive **bonus** points depending on your final ranking on Kaggle's private leaderboard at the end of the competition.

7 Deadlines

- The deadline to beat the baseline is **October 21st, at 23:59**.
- The Kaggle competition will close on **November 9th, at 23:59**.
- You must upload your report and code on Gradescope before **November 12th 23:59**.