Kaggle Competition IFT3395/6390

Instructions for UdeM students

September 30, 2019

1 Background

For this project, you will take part in a Kaggle competition based on text classification. The goal is to design a machine learning algorithm that can automatically sort short texts into a pre-determined set of topics. The dataset that we have prepared contains posts from Reddit. We have selected 20 subreddits: these will be our 20 pre-determined topics (classes). We have sampled 3,500 messages from each subreddit to be used as the training set, and 1,500 messages as test set. You will implement and train a few different classifiers and will be evaluated on the test accuracy that your trained models achieve.

The competition, including the data, is available here: `https://www.kaggle.com/c/ift3395-ift6390-reddit-comments`.

2 Kaggle Team formation

Each team should consist of **2 grad students for IFT6390** or **3 undergrad students for IFT3395**. To form a team:

- Enter the competition and create a Kaggle account if you are not registered yet by following the link: `https://www.kaggle.com/t/79d35aaec8f642a58ef289a52b67bfc7`
- In the "Invite Others" section, enter your teammates' names, or team name.
- Your teammate has the option to accept your merge.
- Fill out the google form `https://forms.gle/VvL28nGP1MMYrCnC7` with your team information by **Oct 11th at 23:59**. Any teams not registered or registered late will not be graded.

Important note: The maximum amount of submissions is 2 per day, per TEAM. Any team whose individual members have a submission count larger than what is allowed up to-date will be UNABLE to form a team. Example: Today is the first day of competition. A,B,C are three teammates who haven't formed a team yet.

- A submitted 0 times.
- B submitted 2 times.
- C submitted 1 time.

Because the maximum amount of submissions is 2 per team per day, the total possible submissions for a team is 2. However, the cumulative submission count for A,B,C is 3. Therefore, they will be unable to form a team (They will need to wait for tomorrow, and not submit any submissions for the next day).

You can start submitting solutions before you form a team, as long as you are careful about the above limitation when forming teams.

3 First milestone: Beat the baselines (Oct 18th)

For this first milestone you will need to build a Naïve Bayes classifier using Bag of Words features, and beat the baselines highlighted in the leaderboard. These baselines are:

- a Random classifier that randomly picks a class for each test example
- a Naïve Bayes classifier using Bag of words features
- a Naïve Bayes classifier with additional Laplace smoothing

To participate in the competition, you must provide a list of predicted outputs for the instances on the Kaggle website. You can submit 2 predictions per day over the course of the competition, so we suggest you start early, allowing yourself enough time to submit multiple times and get a sense of how well you are doing.

You will need to submit to Gradescope the code you used for your Naïve Bayes implementation, including a short Readme document explaining how it is organized before **Oct 18th 23:59**. Part of your overall grade will be attributed according to your best test submission on Kaggle on **Oct 18th 23:59**. For each of the 3 baselines that you beat, you get extra points.

... but the competition is not over ! You now have the opportunity to improve your model and aim for the best possible performance during the second phase.

4 Second milestone: End of competition (Nov 6th)

You have until **Nov 6th 23:59** to achieve the best performance you can on the same task. In this phase you are free to implement any method you think would work best, with the following restrictions.

For this milestone you must try at least **2 other models** than Naïve Bayes, and compare their performances. You are encouraged to implement techniques studied during the course, and look up for other ways to solve this task. Here are a couple of possibilities:

- Kernelized SVM using string kernels

- Random Forests
- Hand-crafted features and logistic regression
- any other algorithm of your choice...

The goal is to design the best performing method as measured by submitting predictions for the test set on Kaggle. Your final performance on Kaggle will count as a criterion for evaluation (see below). If a tested model does not perform well, you can still add it in your report and explain why you think it is not appropriate for this task. This kind of discussion is an important feature that we will be using to evaluate your final competition report.

5 Third milestone: Report (Nov 8th)

In addition to your methods, you must write up a report that details the preprocessing, validation, algorithmic, and optimization techniques, as well as providing results that help you compare different methods/models. The report should contain the following sections and elements. You will lose points for not following these guidelines.

- Project title
- Team name on Kaggle, as well as the list of team members, including their full name and student number.
- Introduction: briefly describe the problem and summarize your approach and results.
- Feature Design: Describe and justify your pre-processing methods, and how you designed and selected your features.
- Algorithms: Give an overview of the learning algorithms used without going into too much detail, unless necessary to understand other details.
- Methodology: Include any decisions about training/validation split, distribution choice for naïve bayes, regularization strategy, any optimization tricks, setting hyper-parameters, etc.
- Results: Present a detailed analysis of your results, including graphs and tables as appropriate. This analysis should be broader than just the Kaggle result: include a short comparison of the most important hyperparameters and all methods (at least 3) you implemented.
- Discussion: Discuss the pros/cons of your approach & methodology and suggest ideas for improvement.
- Statement of Contributions. Briefly describe the contributions of each team member towards each of the components of the project (e.g. defining the problem, developing the methodology, coding the solution, performing the data analysis, writing the report, etc.) At the end of the Statement of Contributions, add the following statement: We hereby state that all the work presented in this report is that of the authors.

- 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 report and your code on Gradescope before **Nov 8th 23:59**.

Submission Instructions

- You must submit the code developed during the project. The code must be well-documented. The code should include a README file containing instructions on how to run the code.
- The prediction file containing your predictions on the test set must be submitted online at the Kaggle website.
- The report in pdf format (written according to the general layout described earlier) and the code should be uploaded on Gradescope.

6 Evaluation Criteria

Marks will be attributed based on the 3 milestones:

1. You will be assigned points for each one of the 3 baselines that you beat. You will only get these points if you submit your implementation of Naïve bayes to Gradescope before the deadline for Milestone 1.
2. You will be assigned points depending on your final performance at the end of the competition, given by your ranking in the private leaderboard.
3. You will be assigned points depending on the quality and technical soundness of your final report (see above).

7 Exact Deadlines

- The deadline for team formation is **October 11th, at 23:59**
- The first milestone (beating the baselines and submitting the code on gradescope) is **October 18th, at 23:59**
- The Kaggle competition will close on **November 6th, at 23:59**
- You should upload your report and code on Gradescope before **November 8th 23:59**