Exercise 3 - Language Identification

Reloaded and Convoluted

Deadlines

The deadline for Exercise 3 is **09.11.2020, 00:15 CET**
The deadline for the peer review is **16.11.2020, 00:00 CET.** You will find instructions for the peer review process at the end of this document.
The deadline for feedback to your peer reviewers is **20.11.2020, 00:00 CET**

Learning goals

This exercise builds on the language identification task you solved in Exercise 1. By completing this exercise you should …

- … understand CNNs.
- … be able to implement CNNs in PyTorch or TensorFlow/Keras
- … deepen your understanding of the role of hyper-parameters, regularisation, and handling class imbalance
- … perform an error analysis of machine learning models.

Please keep in mind that you can always consult and use the exercise forum if you get stuck (note that we have a separate forum for the exercises).

Deliverables

We encourage you to hand in your solutions as a Colab-Notebook. **Download your notebook as a .ipynb file**. That way your reviewers can view and execute your code. Or can view your already executed code.

Please hand in your code and your lab report. Hand in the following files and name them exactly in the following fashion:
-ex03_cnn.ipynb

-ex03_labreport.pdf

zip it and name the zip-folder *ex02_ml4nlp1.zip*.

The .ipynb files contain your well documented AND EXECUTABLE code. We recommend you use Google's Colaboratory, where you have access to GPU time.

If you prefer to solve the exercise on your own computer please submit a zip-folder containing the required files:

- ex03_cnn.ipynb
- ex03_labreport.pdf

We assume that the data files are in the same folder as the scripts, e.g.

- ex03_cnn.ipynb

Please submit the lab report in PDF format. The lab report should contain a detailed description of the approaches you have used to solve this exercise. Please also include results. In this exercise description, we highlight places in green where we expect a statement about an issue in your lab report.

Please note:

- Your peers need to be able to run your code. If it does not work, you will not be able to obtain the maximum number of points.
- DO NOT submit the data files!

Data

For this exercise, you will work with the same data as for Exercise 1. The goal is again to classify the language of Tweets. This is an extension of the problem described in Goldberg, chapter 2. However, we will work with more languages than just six and the text segments we need to classify are much shorter. Inspect the data and see how it is distributed.
Use the same training and test split as you did for Exercise 1. Otherwise, your results will not be comparable.

The material folder in the exercise section of OLAT contains the two files “train_dev_set.tsv” and “test_set.tsv”. The files are also published under these two links:

- train_dev_set.tsv: https://docs.google.com/spreadsheets/d/e/2PACX-1vTOZ2rC82rhNsJduoyKYTsVeH6ukd7Bpxvxn_afOibn3R-eadZGXu82eCU9IRpl4CK_gefEGsYrA_oM/pub?gid=1863430984&single=true&output=tsv
- test_set.tsv: https://docs.google.com/spreadsheets/d/e/2PACX-1vT-KNR9nuYatLkSbzSRgpz6Ku1n4TN4w6kKmFLkA6QJHTfQzmX0puBsLF7PAAQJQAxUpgruDd_RRgK7/pub?gid=417546901&single=true&output=tsv

To make the start a little easier, you can go to this notebook in Google Colab which loads the two files using the public links. If you want to, you can just continue the exercise in your own copy of that notebook. If you choose to work locally, download the two files on your computer.

Part 1 - Language identification with CNN

Implement a language identifier in your desired framework. You can use an OOP style like in Rao and McMahon. You can take their Vocabulary class as guidance, but keep in mind that here the input is not words, so your “vocab” will look differently.

1. Your goal is to find the optimal training regime for your CNN classifier. Try out five different combinations of the following and report them in a table and perform an Ablation study (see below).:
   a. optimizer
   b. learning rate
   c. dropout
   d. # of filters
   e. different strides
   f. different kernel sizes
   g. different pooling strategies
   h. batch sizes

   Report the hyperparameter combination for your best performing model on the test set.

Suggestion - Make use of Early Stopping to keep track of loss. You should use a hyperparameter tuning method/package like Sklearn’s GridSearchCV/TALOS (1,2) to find the best hyperparameter combination for the test set.

Important - 2. Perform a well explained Ablation Study on the 5 different combinations of the hyperparameters above. Explain what you understand from different hyperparameters and its effect on the model’s performance. It’s not necessary to be correct with your ablation study understanding but a poorly written ablation study would not fetch full points.

3. Compare the outputs of the best CNN model to your best performing model from Exercise 1. Which classifier scores higher on the test set? Do you have an idea, why this might be?
4. Use a confusion matrix to do your error analysis and summarise your answers in your report.
5. Plot the Heatmap correlation between the hyperparameters above to understand the dependency of hyperparameters on each other and discuss. It’s a good way to analyze the hyperparameters which affect the model most. (example here)

*You must use an internal development set taken from the training set and do a model selection on the development set. DO NOT use the test set to find the best parameters. Report your test set results in the forum - Exercise 3 - Test Set Scores

Important

Please make sure you run your code on Google Colab with GPU selected. Make the GPU selection from “Edit” → “Notebook Settings” and then make the GPU hardware accelerator.

Peer Review Instructions

First: go to www.eduflow.com/join and join the class with the code **42D2XJ**. Important: Register with the E-mail address you use for OLAT.

As soon as the deadline for handing in the exercise expires you will have time to review the submissions of your peers. You need to do **2 reviews** in order to get the maximum number of points for this exercise.

Here some more rules:

- If you do not submit 2 reviews, the maximum number of points you can achieve is 0.75 (from a total of 1).
- Please use full sentences when giving feedback.
- Be critical, helpful, and fair!
- **All reviews are anonymous: Do not put your name into the python scripts, the lab report or the file names.**
- You must also give your reviewers feedback. The same criteria as above apply.
- Students that consistently provide very helpful feedback can be awarded a bonus in case they earned less than 6 points in total. Ways to obtain points are thus the following:
  - 5 exercises = 5 points

    - 1 presentation or research paper dissection = 1 points
    - consistently good reviews = 1 point

Groups:

- You can create groups of two to solve the exercise together.
- Both students should submit the solutions separately and the same lab report can be submitted. Otherwise, points of team-mates might differ based on feedback on the lab report.
- When submitting the exercise, write a small post in the “Groups”-thread in the exercise forum on OLAT to notify the instructors about the group. **Please notify first hand if you change the group/ decide to work alone.**
- As a group member, you still have to review two submissions with your own eduflow account. However, you may work together in the group to write all 4 reviews.