

# NLP 2024

## Homework 1B instructions

---

Code a classification model!

Slides provided by:

- Roberto Navigli
- Luca Giofrè
- Lu Xu
- Luca Moroni
- Tommaso Bonomo
- Alessandro Scirè



# Rationale

- This is a **mandatory** follow up of HW 1.A to make it “computational”
- Each student will deliver their homework individually
- You cannot share (in any way!) your homework
  - All the students involved **for any reason** in code sharing will see **ALL** their homeworks rejected

# Sentence Classification task

- Sentence Classification task is the task of classifying an entire sentence (or any other span of text) using a predefined set of classes.
- Some common use cases are sentiment analysis and natural language inference.
- You should use all tokens in the sentence to compute a cumulative representation of it that can be used to predict the correct class.

# Baselines



# What is a baseline

Given a task, you can always define a **baseline**, i.e., a trivial approach that solves it in a naïve way.

Baselines act as the foundation for assessing the effectiveness of more sophisticated approaches.

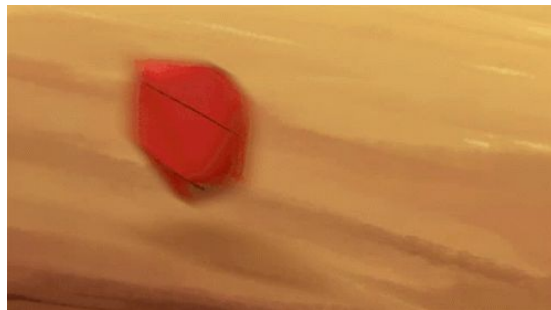
Baseline approaches serve to:

1. **establish a performance standard** against which progress can be assessed
2. offer a **reference point for evaluating** the effectiveness of sophisticated models
3. **spot important imbalance in the given data.**

Let's have a look at the most intuitive baselines for our classification tasks...

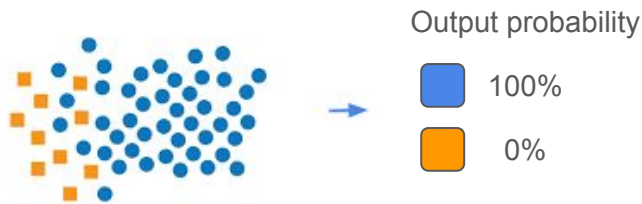
# Random Baseline

- The Random baseline selects a label uniformly at random among the possible labels.
- If your model cannot beat this, there is something wrong with your approach...



# Majority Class Baseline

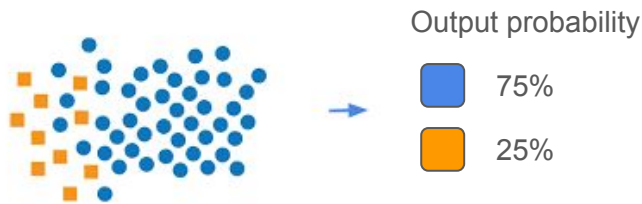
- The Majority Class baseline always predicts the most frequent class in the training set.
- This baseline is suitable to spot imbalanced datasets where the majority class dominate the distribution.



<https://www.linkedin.com/pulse/some-tricks-handling-imbalanced-dataset-image-m-farhan-tandia>

# Stratified Baseline

- The Stratified baseline predicts the classes following the distribution of labels in the training set.
- This baseline also is useful to spot imbalance in the data distribution.



<https://www.linkedin.com/pulse/some-tricks-handling-imbalanced-dataset-image-m-farhan-tandia>



# Implement your Baseline

- The first thing to do in this task (and in general, when you are experimenting) is to choose and set up a trivial baseline.
- **Baselines need to be trivial and are meant to be surpassed.** In this way, you can compare your idea and your work with it (i.e., *is my model better than flipping a weighted coin?*)
- Implementing a simple baseline is a requirement.

# The data



# Subset of original dataset pool

For HW1B, you will use a subset of the datasets of HW1A. Each student will have one of their (previously assigned) dataset to work on ([see this document](#)), already converted to the JSONL format.

- 0-emotivita: only task A
- 1-homotransfobia: only task A
- 5-discotex: only task 2
- 7-AMI: only task 1
- 8-sardistance: only task 1
- 9-haspeede: only task 1
- 18-accomplit: only task 1
- 21-ironITA: only task 1
- 22-GxG: no genre distinction
- 24-sentipolc: only task 1
- 27-haspeede3: only task 1
- 28-pretens: only task 1

# Fallback Dataset

Students lacking any of the selected datasets will be given another dataset complying to the format of HW1A (processed by us).

The *fallback* dataset is HaSpeeDe (only data of task A), which you can download from this [link](#).

# Fallback Dataset

The .jsonl files schema is as follows:

- text: the **sentence** to be classified
- choices: the possible **labels** (“odio” or “*neutrale*”)
- label: the **index** of the correct label

# Fallback Dataset

Folder structure:

- `train-taskA.jsonl`
- `test-news-taskA.jsonl`
- `test-tweets-taskA.jsonl`

You will train your model on the training data and test it on the two different test sets.

# Your implementation

- You will provide the code for training and evaluating an LSTM on your assigned dataset.
- Feel free to take inspiration from Notebook #4 extra, but do not simply copy from it!
  - You can experiment with the choices of modules (RNNs, LSTMs, etc.), the model architecture, the number of hidden dimensions, and with the hyperparameters (i.e., training epochs).
  - You shall consider only the topics up to the BiLSTM to solve this task (i.e., no Transformers, no BERT, no pre-trained models) and cannot use PyTorch Lightning.
  - Please do note that the last point applies **also to the sophisticated baseline** (the extra).

# Submission and Deadline



# Grading

- Each student will be evaluated on their individually assigned dataset.
- We will assign up to 30 points for the task, including extras you can obtain up to 33/30 points:
- We will follow this breakdown:
  - Report validity and adequate explanation of the work done:  $\frac{1}{2}$
  - Code provided and working:  $\frac{1}{4}$
  - Performance of your (best) model:  $\frac{1}{4}$
- **Extra points**
  - +1 for attendance on Friday 19th
  - +2 for doing a comparison with a simpler model (i.e., a custom baseline)

# Extra points

- We will assign **1 point** to each student who will be present in class on Friday, 19th and will start the homework with us.
  - This is a good opportunity to ask questions and get answers right away from your TAs :)
- We will assign **2 points** to each student who will implement another (simpler!) model (e.g., using Word2Vec) and will do a comparison analysis of the performances
  - If you choose to do this extra task, you can use **up to 1 page and a half** (always excluding figures and tables)

# Report

You will write an **individual report** (up to 1 page, excluding tables, figures, and references).  
Be concise.

1. Introduction
2. Description of the dataset (**brief**)
3. Architecture of your model (figures are okay)
4. Design choices of your model (*why did you choose to do x instead of y?*)
5. Baselines implemented (**brief**)
6. Results section (*how does your model compare to the baselines?*)
7. Instructions to run your code (**unambiguous**)
8. Any other information you think may be useful for us

# Report Format

- Use **Latex** to write the report. If you don't know how to use it, it is a good time to start learning :)
- I suggest you to write your report using [Overleaf](#) for the easiest possible experience
- Use the current ACL template (available as [Overleaf](#) template or in [GitHub](#))
- Each report should consist of **up to 1 page**, excluding tables, figures and references (which should be at the end of the report).

# Deadline

- Delivery deadline: 23:59ish Tuesday April 30th Italian time (CET)
- Late submission penalty: we will deduct **1 point for each day** after the deadline, up to a maximum of -5 points.

No submissions are accepted beyond 23:59 Sunday May 5th (CET)

# What to deliver – Example of folder structure

Example of folder structure:

- **HM1\_B-<student\_id>/**
  - hw1b\_report.pdf *# your 1-page and a half report*
  - hw1b\_train.py *# this launches the training step*
  - hw1b\_evaluate.py *# this launches the evaluation on the test set*
  - **src/** *# put here all the rest of your code, if needed*
  - **data/** *# put here the assigned data*

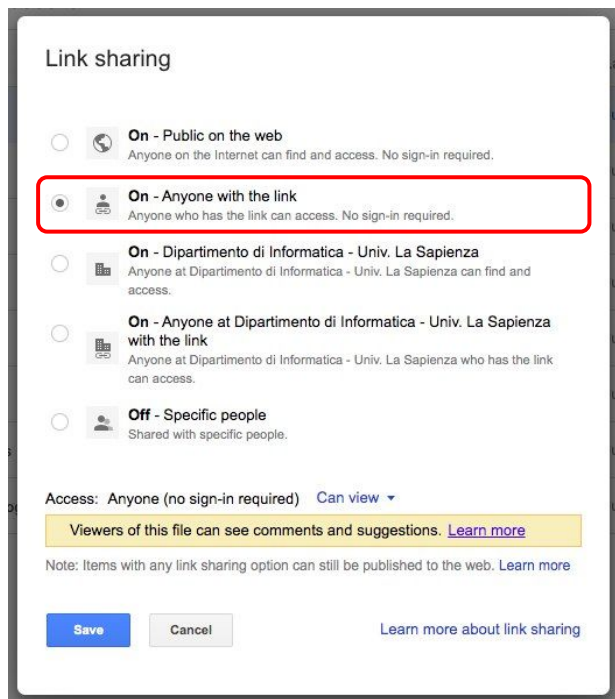
If you want, you are free to provide your code in a notebook, as long as it works and it is easily runnable with the 'Run all' button.

# What to deliver – Example of folder structure

Note that you cannot in any way modify the test set of the dataset assigned to you.


Keep in mind that, after you submit, we will evaluate your model on the original test set.


# Submission Instructions





The screenshot shows the 'Link sharing' settings for a Google Drive file. The 'On - Anyone with the link' option is selected and highlighted with a red rectangle. Below the options, the 'Access' is set to 'Anyone (no sign-in required)' with a 'Can view' dropdown. A yellow banner states 'Viewers of this file can see comments and suggestions. [Learn more](#)'. A note at the bottom says 'Note: Items with any link sharing option can still be published to the web. [Learn more](#)'. At the bottom are 'Save' and 'Cancel' buttons, and a link to 'Learn more about link sharing'.


Link sharing

☐  **On - Public on the web**  
Anyone on the Internet can find and access. No sign-in required.

☒  **On - Anyone with the link**  
Anyone who has the link can access. No sign-in required.

☐  **On - Dipartimento di Informatica - Univ. La Sapienza**  
Anyone at Dipartimento di Informatica - Univ. La Sapienza can find and access.

☐  **On - Anyone at Dipartimento di Informatica - Univ. La Sapienza with the link**  
Anyone at Dipartimento di Informatica - Univ. La Sapienza who has the link can access.

☐  **Off - Specific people**  
Shared with specific people.

Access: Anyone (no sign-in required) [Can view](#) ▼

Viewers of this file can see comments and suggestions. [Learn more](#)

Note: Items with any link sharing option can still be published to the web. [Learn more](#)

[Save](#) [Cancel](#) [Learn more about link sharing](#)

- Upload the zip on your **institutional** Drive and make it **link-shareable** and **public** to anyone (an automatic script will download it).
- Make sure it is accessible via an incognito page of your browser!
- You have to submit the homework through the [submission form](#) on Google Classroom. You will be asked to fill a form with the requested information and the **link** to the zip you uploaded on Drive.



# Plagiarism

Collaboration among students is **not** allowed.

We will check for plagiarism both manually and automatically.

It is **not** allowed to:

- Copy from other students.
- Share your code across students.
- Use ChatGPT or similar systems **for report writing**

Projects under any of the above conditions will be desk-rejected

# Any doubts? (hopefully not)

Use the Classroom group if you have any questions. Only after you cannot solve your issues in the group, write to **ALL the TAs**, so to have a higher reply rate ;)

- Luca Moroni: [moroni@diag.uniroma1.it](mailto:moroni@diag.uniroma1.it)
- Luca Giofrè: [gioffre@diag.uniroma1.it](mailto:gioffre@diag.uniroma1.it)
- Lu Xu: [xu@diag.uniroma1.it](mailto:xu@diag.uniroma1.it)
- Tommaso Bonomo: [bonomo@diag.uniroma1.it](mailto:bonomo@diag.uniroma1.it)
- Alessandro Scirè: [scire@diag.uniroma1.it](mailto:scire@diag.uniroma1.it)

Start the mail subjects with “[NLP 2024 HW1.B]”

