Text Detoxification

- Please read the document carefully and follow instructions. If you have questions ask in the telegram group.
- Use this template to complete your assignment report and upload it to Canvas:

  https://colab.research.google.com/drive/1ttPT6X4K0ovgbzmNjlcEiprkj1LaBuF2
- Submit your results to Codalab (post evaluation phrase):

  https://codalab.lisn.upsaclay.fr/competitions/642

  - **Important: As the name of the team in Codalab indicate 'DL4NLP-23'.**
  - Important: In the Jupyter notebook indicate name of your Codalab user.

Introduction

Global access to the Internet has enabled the spread of information all over the world and has given many new possibilities. On the other hand, alongside the advantages, the exponential and uncontrolled growth of user-generated content on the Internet has also facilitated the spread of toxicity and hate speech. Much work has been done in the direction of offensive speech detection. However, it has become essential not only to detect toxic content but also to combat it in smarter ways. While some social networks block sensitive content, another solution can be to detect toxicity in a text which is being typed in and offer a user a non-offensive version of this text. This task can be considered a style transfer task, where the source style is toxic, and the target style is neutral/non-toxic. The task of style transfer is the task of transforming a text so that its content and the majority of properties stay the same, and one particular attribute (style) changes. This attribute can be the sentiment, the presence of bias, the degree of formality, etc. Considering the task of detoxification, it has already been tackled by different groups of researchers, as well as a similar task of transforming text to a more polite form. However, all these works deal only with the English language. As for Russian, the methods of text style transfer and text detoxification have not been explored before

Task formulation

You have a great chance to be the first participant in the competition of automatic detoxification of Russian texts to combat offensive language. Such a kind of textual style transfer can be used, for instance, for processing toxic content in social media. While much

work has been done for the English language in this field, it has never been solved for the Russian language yet.

We define the detoxification task as the task of style transfer: from the toxic style to the non-toxic style. We want to rewrite the sentence and preserve the context.

We define the task of style transfer as follows. Let us consider two corpora $D^{\mathrm{X}} = \{x_1, x_2, \ldots, x_n\}$ and $D^{Y} = \{y_1, y_2, \ldots, y_m\}$ in two styles –s^{X} (toxic) and s^{Y} (non-toxic), correspondingly. The task is to create a model $f_\theta : X \to Y$, where X and Y are all possible texts in styles s^{X} and s^{Y}. The task of selecting the optimal set of parameters θ for f consists maximising the probability $p(y' | x, s^{Y})$ of transferring a sentence x in style s^{X} to the sentence y' which saves the content of x and has the style s^{Y}. The parameters are maximized on the corpora D^{X} and D^{Y} which can be parallel or non-parallel. We focus on the transfer $s^{\mathrm{X}} \to s^{Y}$, where s^{X} is the toxic style, and s^{Y} is neutral.

Evaluation metrics

To perform a comprehensive evaluation of a style transfer model, we need to make sure that it (i) changes the text style, (ii) preserves the content, and (iii) yields a grammatical sentence. The majority of works on style transfer use individual metrics to evaluate the three parameters. In our competition we use the following metrics:

1) Joint metric J which combines automatic Style transfer accuracy, Meaning preservation score, Fluency score.

2) ChF: character n-gram F score between the rewritten sentence and the manually.

Methods

In the context of this assignment, you will solve a style transfer task on the dataset of comparative sentences provided by the course team. You will need to train a model and submit your solution to the CodaLab competition:
https://codalab.lisn.upsaclay.fr/competitions/642.

You are free to use any methods and/or models for style transfer or pretrained models for text generation (GPT, T5, BART, etc.). A good baseline solution is to train a transformer decoder or encoder-decoder model on the provided parallel data. However, a lot of students and past participants used this solution, so to improve upon it you may try using more training data and/or change the model structure and/or change the loss function of the model.

You can read more about various solutions in here https://aclanthology.org/2022.acl-long.469/ and here https://arxiv.org/abs/2109.08914 and papers mentioned in the related work section. Also check the lecture on text detoxification delivered in the beginning of the class to get more ideas on the methods you can use.

Expected output

Example of the input and model output are presented below:

Model	Sentence
Input	не дай бог моя дочь так оденется убью н[redacted]й палкой (If, God forbid, my daughter goes out dressed like this, I'll f[redacted]g kill her with a stick)
Delete	не дай бог моя дочь так оденется убью палкой (If, God forbid, my daughter goes out dressed like this, I'll kill her with a stick)
Retrieve	не бросайте угла родного одной мы лежали больнице палате в в в те девчонкой была молодой годы (don't abandon your native corner same hospital we were ward in in in those girl was young years)

As output we expect a paraphrased (rewritten) toxic sentence in a more neutral (nontoxic) style. For each input sentence x_i we expect a corresponding rewritten sentence y_i. Please, submit a textual file results.txt each rewritten sentence in a row. Make sure that the size of the output dataset is the same as the size of the input dataset. Please, submit the result file in the zip archive in Codalab.

You are expected to:

1. Develop a solution of the task and provide a reproducible code in the form of an ipython notebook (preferably a link to a Google Colab[3] notebook):
○ You should use Python 3,

○ The notebook should contain code for installation of all dependencies,

○ The notebook should contain code for downloading all the additional datasets it uses,

○ The notebook should reproduce the results you submit to CodaLab:

i. It should generate the output file of the required format and compute the scores for it - this result should be reached by running your notebook cell by cell without changing anything, including paths to files!

ii. The scores should be close to the scores of your CodaLab submission.

iii. If this reproducibility is not reached your grade will be lowered.

2. Write a report which describes the method used in your solution as a part of your jupyter notebook.

3. Push the (best) solutions which you developed to the CodaLab competition so that they appear in the leaderboard. The name of your user / submission should be present in the report for verification.