Prediction of message toxicity on the Internet
Machine Learning for Natural Language Processing 2020

Corentin ODIC
ENSAE
corentin.odic@ensae.fr

Michaël SOUMM
ENSAE
michael.soumm@ensae.fr

Abstract

In this work, we train a classifier based on BERT in order to predict whether an Internet comment is toxic or not. After carefully constructing a balanced dataset, we reach an accuracy of 86%. We then try to predict the type of toxicity, but with less convincing results. Our .ipynb notebook is available at Github[1] or Colab[2].

1 Problem Framing

Whether it is for moderating Internet social networks or for training NLP models in bases that respect human decency, companies may want to empty their bases of so-called "toxic" messages. However, this problem can be quite technical and can lead to a bias when an often denigrated community is mentioned in a post. Indeed, messages about homosexuals, women or black people are often associated with insults, but we should not predict toxicity as soon as one of these communities is evoked.

We will use a database[3] provided by "the Civil Comments", a platform that wanted to automate the moderation of violent messages on the internet and that made its data public after its closure in 2017. It contains 2'000'000 sentences of which 90% are annotated with a toxicity score and a specification of the source of the problem (racial, misogynistic, homophobic,...).

Our goal will be to train a classifier on this database in order to predict whether a comment is toxic or not, and then try to fine-tune it in order to predict the type of toxicity.

[1] https://github.com/MSoumm/ENSAE-3A-NLP/blob/main/NLP_Project_ODIC_SOUMM.ipynb
[2] https://colab.research.google.com/drive/1KaSoZZyax0vP-Hm4aMPu-oEk5IDbaaKR
[3] https://www.kaggle.com/c/jigsaw-unintended-bias-in-toxicity-classification/data

2 Experiments Protocol

2.1 Exploration of the database

For each comment, we use 2 types of features :

- `target` : a toxicity score in $[0, 1]$
- `Obscene, Insult, Threat, Identity Attack` : toxicity type scores in $[0, 1]$

As a baseline, a comment is considered toxic if it has a toxicity score higher than 0.5.

We need to first establish that a simple pretrained model, such as NLTK's *Sentiment Analysis* is not sufficient to predict the toxicity. After dividing the database into toxicity levels, we plot the predicted negativity distribution for each level (see Figure 2), and see that a more complex model will be needed.

2.2 Toxicity score

We first focus only on the main toxicity score. Since the dataset is too large for a reasonable training time and highly imbalanced, we chose to draw a more equilibrated sample : 25% of non toxic, 25% of mildly toxic $(0 - 0.5)$ and 50% of toxic $(0.5 - 1)$, in order to get 50'000 comments. This construction may be debatable, but we came to the conclusion that it will lead to a good diversity in our custom database, and to have a distribution that allows for a good interpretation of accuracy results. Furthermore, we choose a maximum length of 128 tokens which conserves most comments while being computationally acceptable.

After standard preprocessing, we train a classifier based on pretrained BERT with custom final feed-forward layers with dropout. Several trainings were tried :

- Train BERT + 1 final layer
- Freeze BERT and train 2 final layers

- Train only BERT final block + 1 final layer

In each case, the learning rate is 10^{-3} for the final layers(s) and 10^{-5} for the BERT embeddings, with an Adam optimizer. We use a binary cross-entropy loss and monitor training by plotting the accuracy and loss.

2.3 Toxicity type

The 4 toxicity types are also quite imbalanced, not only within each class but also with respect to each other. We under-sample some classes and under-sample others in order to have a more balanced dataset. This is quite technical since we have a multi-label problem.

We will use the trained model on the toxicity, and only retrain the final layer, since the embeddings should already be adapted to this problem. We use the same loss and optimizer as before.

3 Results

3.1 Toxicity

The best results were found by training the whole model with BERT, even if it overfitted after only 3 epochs (Figure 1). Training only the last layers lead to a chaotic training and low accuracy. Training the final blocks of BERT lead to overfitting but with less accuracy. We achieve an accuracy of over

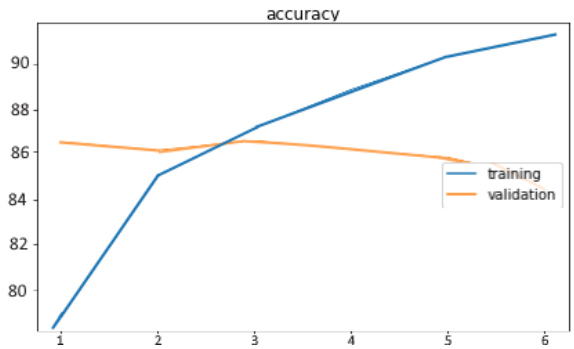

Figure 1: Training of selected model

86%, with the following confusion matrix:

	N	P
N	3707	476
P	611	3723

Table 1: Confusion matrix of the model on the validation set

which corresponds to a precision of 89% and a recall of 86%.

By printing some of the predicted scores (see notebook), a qualitative evaluation leads to quite convincing results.

3.2 Toxicity type

We monitor the training by plotting the accuracy for each class (see Figure 3). We don't seem to have any overfit and the model reached good accuracies after 20 epochs. Let us look at some metics for each class. As we see, even if the model

	precision	recall	f1-score	support
obscene	0.77	0.36	0.49	845
identity_attack	0.92	0.29	0.45	1186
insult	0.88	0.86	0.87	3178
threat	0.92	0.39	0.55	1473

Table 2: Metrics for each toxicity type

has a good precision for all classes, it only has a good recall for the `insult` type, which is the most represented one. Therefore, we have a bias that we could not solve by constructing a custom database.

This biais can be seen when printing predictions for each sentence (see notebook).

4 Discussion/Conclusion

In this work, we trained successively 2 models to predict the toxicity and the toxicity type of internet comments. Even if we acheived good results with the first one, through a well-constructed database and an adapted training, the results for the second one are less convincing. This comes mainly from the dataset imbalance, which could be solved through more advanced sampling techniques such as SMOTE for multi-label problems.

References

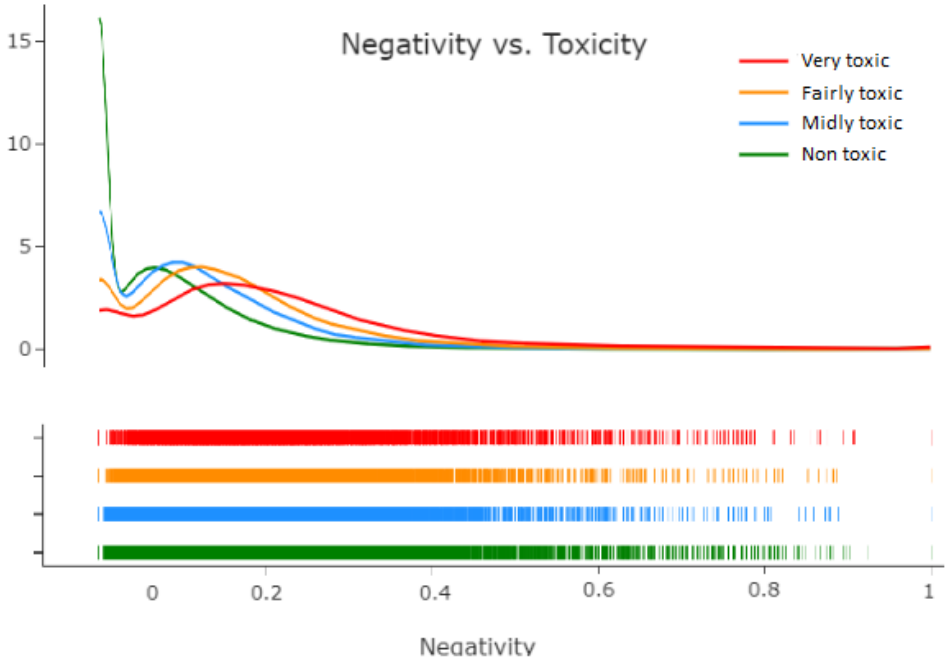

Figure 2: Negativity for multiple toxicity levels

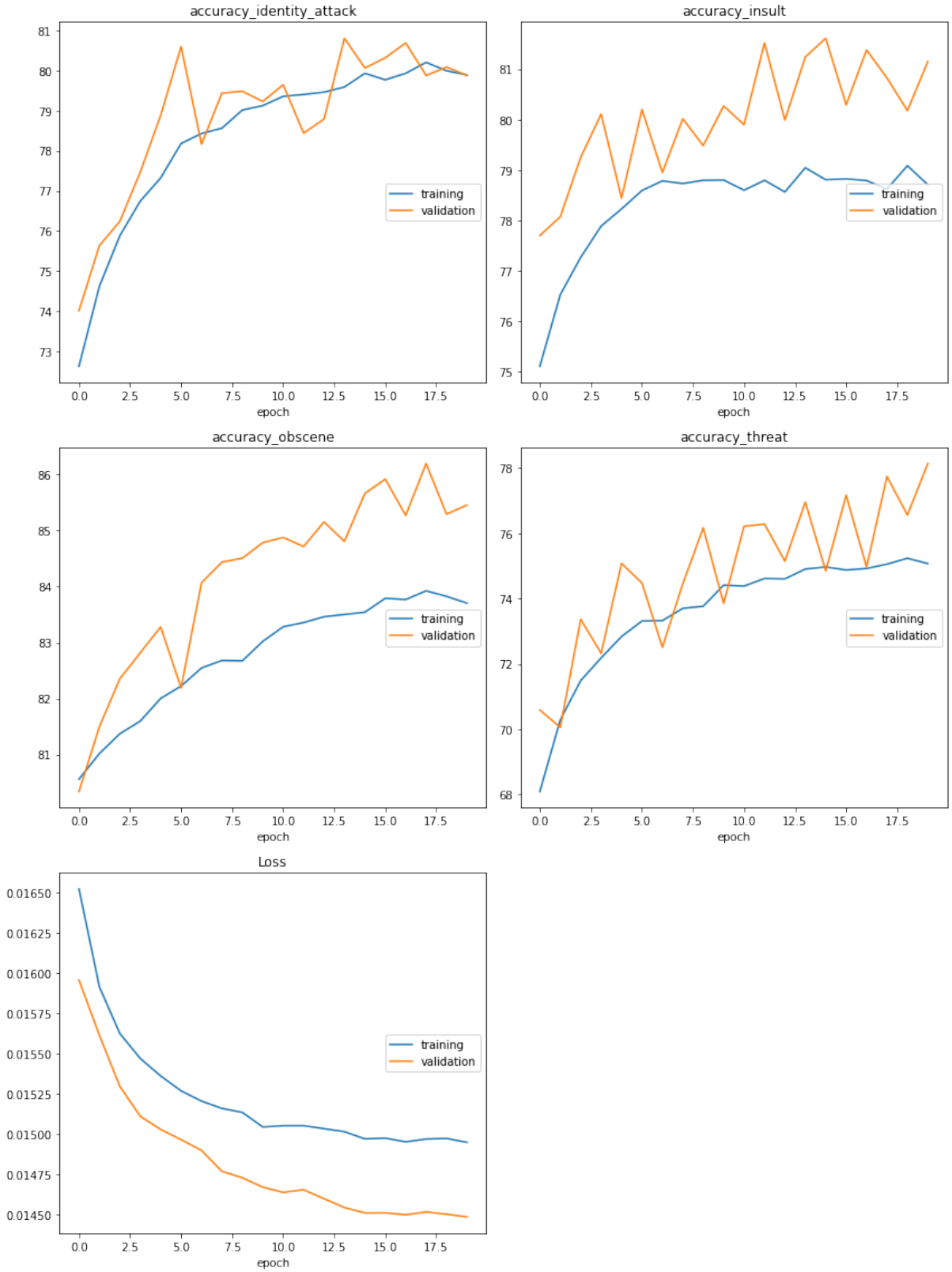

Figure 3: Training of model on the toxicity types