Optimizing RoBERTa for Detection of Patronising and Condescending Language in Media *

Moritz Hauschulz
Imperial College London
meh23@ic.ac.uk

Github: https://github.com/moritzhauschulz/nlp_submission (31707b21951445aa172442696770637fbdc15a4f)

Abstract

This report presents an adapted RoBERTa model to detect the presence of patronising and condescending language (PCL) in text. Thereby, the report addresses subtask 1 of task 4 of the SemEval 2022 competition. We first describe the 'Don't Patronize Me!' dataset (Perez Almendros et al., 2020) and assess the task. We then describe sampling techniques and data augmentation that we apply to improve the baseline model. We assess quantitatively how the different adaptations impact our model and justify the choice of hyperparameters. Our final model achieves an F-1 score of 0.52 on the official validation set. Finally we propose directions for further improvement.

1 Introduction

Patronising and Condescending Language (PCL) is language that indirectly expresses the superiority of the author over vulnerable communities, such as immigrants, low-income households or people with a disability and can cause harm despite potentially benevolent intentions (). Harm may result for instance from exacerbating the exclusion of the affected groups from society by reinforcing stereotypes and biases in the population. This is especially significant when such language is used by large media outlets. The use of compassionate formulations may further hide implicit discrimination, which risks perpetuating the use of such phrasing even further. Due to the subtlety of the characteristics of PCL, classifying text as PCL poses a significant challenge even to humans. The understanding of PCL can vary across individuals, for example since different words may carry different connotations across countries, even within the same language. As a consequence, the identification of PCL is a challenging task for computers, since they learn markers of PCL from human-labelled training data.

1.1 Task and Data

This report is a response to subtask 1 of task 4 of the SemEval 2022 competition, which states that 'given a paragraph, a system must predict whether or not it contains any form of PCL.' This is to be achieved by exploiting the 'Don't Patronize Me!' dataset, which contains paragraphs published in media across 20 English speaking countries. Each paragraph is linked to a manually annotated label with integer values ranging from 0 (no PCL) to 4 (serious PCL), the country in which it was published as well as one of 10 key-words identifying the affected vulnerable community (e.g. poor-families). For the purpose of binary classification, the 'original labels' are collapsed to binary labels, equal to 1 if the original label is 2 or greater and 0 otherwise. The dataset (excluding a held out test set) has has 10469 samples. The ratio between positive and negative labels is roughly 10:1, with only 993 positive labels. Paragraph length is also highly variable around a mean of 48 words, and the longest paragraph containing 909 words. The positive label correlates positively with the paragraph length 1. This is intuitive, since the likelihood of a random string of words containing at least one phrase of condescending language increases with its length.

label	counts	avg_word_count
0	9476	47.88
1	993	53.62

Table 1: Counts and Average Words by Label

While the dataset is roughly balanced between the different communities, the positive rate varies

drastically, from around 0.03 for paragraphs referring to 'migrant' communities to 0.17 in paragraphs referring to 'homeless' communities 8. Thus, if PCL differs significantly across communities, this could result in the model performing poorly for text referring to communities for which few PCL examples were present in the training data. Similarly, the number of examples are roughly balanced across countries, however positive rates vary between 0.6 for Hong Kong and 0.13 for Nigeria.

Identifying PCL is a highly nuanced task, as the understanding of PCL may very across individuals and cultures. 6 shows example paragraphs ranked according to their original label, i.e. in increasing severety of PCL. While the examples corresponding to 0 and 1 are arguably more neutral, the ranking between the examples for labels 2, 3 and 4 seem somewhat arbitrary. While this paper primarily considers the collapsed binary labels, it is important to note that the the boundary between 0 and 1 in this collapsed label may be similarly blurry.

We split this dataset into train and test proportion 80:20 official training and validation set, along the suggested split by the competition organisers. A held-out test set exists, but due to the absence of labels, this was excluded from the analysis above. The thus obtained official train set is again split into an internal training set of size $N_{train} = 6700$ and validation set $N_{val} = 1675$ (80:20). The official validation set (our test set) $N_{train} = 2094$, and will only be considered for the evaluation of our final model.

2 Model

In developing our model, we first considered a few simple baseline set-ups. We then considered how the RoBERTa model can be optimized through sampling techniques, data augmentation and supplementing categorical variables as tokens. Note that we refrain from augmenting the model architecture and demonstrate that substantial performance improvements can be achieved by careful data engineering. Except when stated otherwise, the experimental models are trained on our internal train set and evaluated on our internal validation set. All RoBERTa based models (except the competition baseline) are trained over 3 epochs, evaluating the model on the internal validation set every 50 iterations and finally retaining the highest-scoring model in terms of F1. This is designed to avoid training instability, as well as overfitting to the internal training set. We use an uncased RoBERTa model from Huggingface with a linear learning rate scheduler with warmup, and a learning rate of 4e-5.

2.1 Baseline

To put our proposed model into context, we first provide a baseline logistic regression classifier and a baseline Naive Bayes Classifier both based on BoW features, as well as a RoBERTa (Liu et al., 2019) based sequence classification baseline from the competition. The Naive Bayes model performs poorly, with an F1 score of 0.02. The low recall shows that most examples of the positive class were misclassified. Note that this is despite Laplace smoothing being enabled. The logistic classifier model, also based on BoW features performs markedly better, achieving an F1 score of 0.32. This is likely due to its ability to assign importance to certain keywords for predicting the positive class. However, a recall of 0.22 still suggests a high rate of false negatives. Consider the paragraph 'Very often, the people who are most in need may not read the newspapers [...]. They need people who can talk to them in their language, people who will knock on their doors, check on them to see whether they are okay, and explain some of these assistance schemes to them.', which is labelled as PCL in the training data but misclassified by the logistic classifier. While the presence of certain phrases, such as 'their language' or 'may not read the newspapers' are likely the indicators that motivated the label as PCL, they only amount to PCL when seen in context. A BoW model does not allow for context, and the logistic classifier is designed to sum over the contribution of each of the words regardless of order. As a result, such multi-word contexts cannot be captured. Such contexts are more accurately captured by large pre-trained models such as RoBERTa. Indeed, we see that the simple competition baseline improves markedly on the logistic BoW classifier to achieve an F1 score of 0.52. Note however, that this baseline involved some downsampling on the negative class. Our experimentation has shown that without downsampling all examples in the internal training data are classified as non-PCL. This suggests that the RoBERTa model is susceptible to careful engineering of the training data, as the next

sections will demonstrate.

Table 2: Performance Metrics for BoW and RoBERTa Models

Metric	BoW	Naive Bayes	RoBERTa
F1 Score	0.32	0.02	0.52
Accuracy	0.91	0.90	0.86
Recall	0.22	0.01	0.74
Precision	0.59	0.33	0.40

Table 3: Baseline BoW, NB and RoBERTa

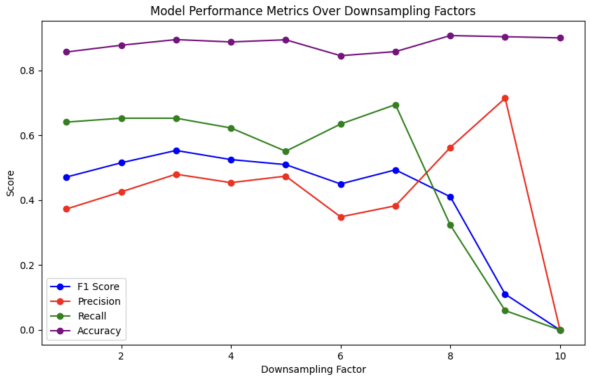

Figure 1: Evaluation Metrics for Different Downsampling Factors

2.2 Sampling

We now turn to augmenting the RoBERTa baseline through sampling techniques. To this end, we consider upsampling the minority (positive) class and downsampling the majority (negative) class.

2.2.1 Downsampling

If N_{pos} is the number of examples of the positive class in our data, then we remove all but $k \times N_{pos}$ examples of the negative class from the training data, where k_{down} is a hyper parameter dictating the ratio between positive and negative samples in the sampled training data. We observe a maximum F1 score at $k_{down} = 3$, which is trending lower for higher values of k and finally falls to zero when $k_{down} = 10$, which corresponds to the case where the full dataset is used. Interestingly, the peak for precision is achieved for $k_{down} = 9$. The upward trend in precision is likely due to the model gaining a better understanding of the characteristics of the negative class, as the number of negative examples increases in the training data. The best model with downsampling is achieved for $k_{down} = 3$ with an F1 of 0.55. 1.

2.2.2 Upsampling

In Upsampling, if N_{pos} is the original number of positive examples in the training data, we re-use each sample k_{up} times. This balances the number of positive and negative samples in the training data, however it introduces duplication of examples. This risks overfitting to the training data. The benefit of upsampling in this way is to increase the importance that the positive examples carry in the training loss, without removing diversity from the set of negative samples (as was the case with downsampling). This can be achieved alternatively through a weighted loss function. We observe that indeed upsampling to $k_{up} = 2$ (duplication of each positive example) has a beneficial effect on all metrics. F1 increases slighlty up to $k_{down} = 8$. The best model with upsampling is achieved for $k_{down} = 8$ with an F1 of 0.58 on the internal validation set. This is higher than the best model for downsampling, likely due to the fact that no information is lost by removing samples from the training data 2.

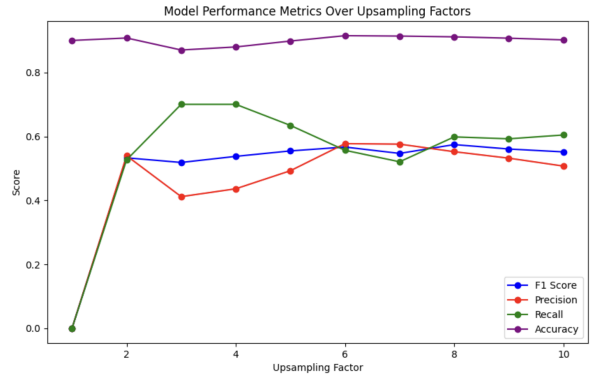

Figure 2: Evaluation Metrics for Different Upsampling Factors

2.3 Data Augmentation

We now consider augmenting the RoBERTa model through data augmentation techniques. As with the sampling techniques above, this is aimed at increasing the balance in the dataset, this time by creating artificial examples from original ones. As stated above, the data is heavily imbalanced towards the negative class, which is why we only consider data augmentation on the positive class. We consider synonym replacement () and back-translation ().

2.3.1 Synonym Replacement

In synonym replacement, new samples are created from existing ones by replacing words with synonymous words. In this case, we rely on synonyms from the lexical database Wordnet (Fellbaum, 2005), which assigns nouns, verbs, adjectives and adverbs to so-called *synsets* of similar meaning. To assess the most effective way of synonym replacement, we consider first replacing nouns, verbs, adjectives and adverbs separately. This is achieved through part-of-speech (POS) tagging. For each real example of the positive class, we create one artificial example with all instances of the respective POS replaced by the first element of its synset. We compare performance to the upsampled model with $k_{up} = 2$, since this data exhibits the same ration of positive and negative samples. We find that noun and adverb replacement improve F1 score significantly, and verb and adjective replacement yields only a small improvement. Our best model with synonym replacement is obtained by replacing nouns only, achieving an F1 score of 0.57 on the validation set. We note that if one is mainly interested in avoiding false negatives, then VERB replacement is a viable alternative, achieving an improved recall of 0.59.

2.3.2 Back-translation

Instead of replacing individual words, back-translation seeks to preserve context by translating the sentences to another language and back. We consider English-French-English back-translation, and use the OPUS-MT pre-trained model (Tiedemann et al., 2023). Again, we compare to the upsampled baseline with $k_{up} = 2$. The trained model scores high on precision (0.61) and relatively low on recall (0.47), with an overall F1 of 0.54. Thus, it provides only a small improvement on the simple upsampling model.

2.4 Categorical Conditioning

We lastly consider conditioning our model on the categorical 'community' variable by augmenting the tokenizer by one additional token per category. For instance, for the 'homeless' label, we add a [HOMELESS] token to the tokenizer. We then prepend the categorical token variable to the text input, yielding text of the form '[HOMELESS] Bauer also suggested the rise in...' for each paragraph. It is important to perform this augmentation on both training and validation data. Note that we adhere to our strategy of not altering the model architecture, albeit in this case modifying the tokenizer to recognize the categorical tokens as units. To allow comparison to previous model augmentation techniques, we upsample the positive class by a factor of 2. We observe only a small improvement in performance compared to the simple upsampling baseline, at an F1 of 0.54.

2.5 Final Model

We compose our final model by combining the different approaches that enabled performance improvement. Note that the main result from the upsampling exercise was that the maximum F1 is reached when upsampling positive examples $k_{up} = 8$ times. Our final model therefore combines all types of data augmentation, yielding five artificial examples per real example and then upsamples the real example an additional two times (factor 3) to yield an 8-fold increase in the number of positive examples in the final augmented dataset. Categorical exclusion reduces performance, so we exclude it 7. Finally, we perform hyperparameter search on the learning rate (in $(1e-4, 1e-6)$, the use of learning rate scheduler (in $\{constant,\ linear\ with\ warmup\}$) and the use of a cased model (in $\{True,\ False\}$). The search did not yield a meaningful increase in F1 score over 30 runs, suggesting that our initial parameter choice is robust. Finally, after determining the best model and parameters through evaluation on the internal validation set, we evaluate the same model on the official validation set (used here as test set). The final model (without categorical inclusion) yields an F1 score of 0.52 on the test set 5. This is an improvement on the RoBERTa baseline model, however we observe a significant drop in performance compared to the internal validation set. This is likely due to the fact that the final model was selected based on F1 performance on the internal validation set.

3 Analysis

In the following, we provide a brief analysis of the model performance on the official validation set (i.e. our test set) by considering the below questions.

Metric	upsampling $k_{up} = 2$	adjective	verb	noun	adverb	back-translation	categorical
F1 Score	0.53	0.54	0.54	0.57	0.56	0.54	0.54
Accuracy	0.91	0.90	0.90	0.92	0.92	0.92	0.91
Recall	0.53	0.59	0.58	0.53	0.53	0.47	0.55
Precision	0.54	0.49	0.50	0.61	0.58	0.61	0.53

Table 4: Upsampling Baseline, Synonyms by POS and Back-translation

Metric	Comp. RoBERTa	Final w/o Categorical Inclusion
F1 Score	0.497	0.522
Accuracy	0.857	0.904
Recall	0.744	0.553
Precision	0.374	0.494

Table 5: RoBERTa Base and Final Model

3.1 To what extent is the model better at predicting examples with a higher level of patronising content?

As expected, the models performance improves with the severity of the PCL present in a given paragraph, as indicated by the original label. Specifically, our model achieves a low recall of 0.22 on paragraphs with original label 2, while attain a recall of 0.69 when conditioning on the highest original label 4 9.

3.2 How does the length of the input sequence impact the model performance?

The accuracy of predictions on the official validation set (our test set) decreases slightly with increasing word count, from around 0.9 for paragraphs of length 30 to around 0.8 for paragraphs of length 140. However, it should be noted that the word count distribution is concentrated below 100, with only few paragraphs exceeding this length. 3.

3.3 To what extent does model performance depend on the data categories?

We note that the model varies significantly across classes. For instance, conditional on community 'migrant' the recall is at around 0.31, while the recall for 'in-need' is at 0.76. Recall from 8, that indeed the 'migrant' class had a very low positive-rate in the training data (0.03), while the positive rate in the 'in-need' class is around 0.16. This lack of diversity in the training samples is a probable cause for the reduced performance on the official validation (i.e. text) data 10.

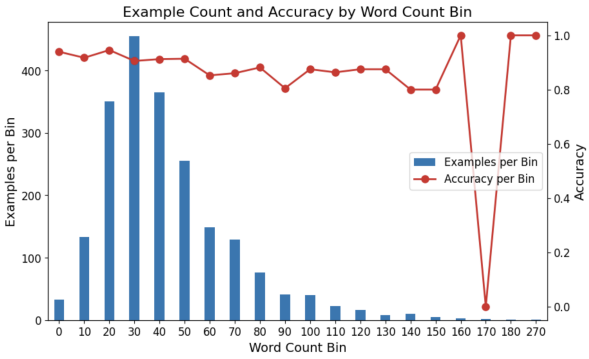

Figure 3: Evaluation Metrics for Different Downsampling Factors

4 Conclusion

We have demonstrated the importance of tackling data imbalance for improving text classification performance with the RoBERTa model. Specifically, we have shown that upsampling is an important factor in increasing performance, and that further improvements can be achieved through data augmentation. However, we observe heterogeneous performance across categories, i.e. communities. For further improvement, upsampling conditional on the positive rate in each category should be considered to ensure that the training data is not only more balanced between positive and negative samples overall, but also categories present in the data. In addition, measures to achieve better generalization performance between validation and test set should be explored.

References

Christiane Fellbaum. 2005. Wordnet and wordnets. In Alex Barber, editor, *Encyclopedia of Language and Linguistics*, pages 2–665. Elsevier.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach.

Carla Perez Almendros, Luis Espinosa Anke, and Steven Schockaert. 2020. Don't patronize me! an annotated dataset with patronizing and condescending language towards vulnerable communities. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 5891–5902, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Jörg Tiedemann, Mikko Aulamo, Daria Bakshandaeva, Michele Boggia, Stig-Arne Grönroos, Tommi Nieminen, Alessandro Raganato, Yves Scherrer, Raul Vazquez, and Sami Virpioja. 2023. Democratizing neural machine translation with opus-mt.

5 Appendix

orig_label	text
0	FLOOD. Two weeks ago, homes and farms were submerged by water, rendering a number of families in Garissa and Tana River homeless.
1	What causes someone to become homeless? Brainstorm as many reasons as you can and check the story for ideas.
2	'Every year I try to get somebody to help me to give some food to some homeless people', he continued.
3	From sleeping rough to becoming a published author: This man's story will make you think twice about ignoring the homeless
4	Hojjat Gharibian was one of hundreds of homeless Iranian survivors, who was huddled against the cold with his family in Qasr-e Shirin.

Table 6: Example Paragraphs Referring to Community 'Homeless'

Metric	upsampling $k_{up} = 8$	Final w/o Categorical Inclusion	Final w/ Categorical Inclusion
F1 Score	0.575	0.581	0.570
Accuracy	0.912	0.913	0.915
Recall	0.599	0.605	0.563
Precision	0.553	0.558	0.577

Table 7: Upsampling Baseline, Synonyms by POS and Back-translation

community	**count**	**positive_rate**
hopeless	1005	0.12
migrant	1089	0.03
immigrant	1061	0.03
disabled	1028	0.08
refugee	1068	0.08
in-need	1082	0.16
homeless	1077	0.17
vulnerable	1080	0.07
women	1070	0.05
poor-families	909	0.17

Table 8: Statistics of Communities and Their Respective Counts and Positive Rates

Original Label	**2**	**3**	**4**
Recall	0.22	0.48	0.69

Table 9: Recall Scores by Original Label

Table 10: Recall Values for Different Communities

Community	Recall
Hopeless	0.51
Migrant	0.31
Immigrant	0.33
Disabled	0.48
Refugee	0.47
In-need	0.76
Homeless	0.41
Vulnerable	0.53
Women	0.4
Poor Families	0.51