

# Imperial College: ‘Don’t Patronize Me!’ Coursework 2024

Anton Zhitomirsky

Imperial College London

az620@ic.ac.uk

[gitlab.doc.ic.ac.uk/az620/nlp-cw-2024](https://gitlab.doc.ic.ac.uk/az620/nlp-cw-2024)

## 1 Introduction

Most social media websites contain portals through which even unregistered users can view their content. Focusing primarily on textual posts, this leaves many vulnerable users to unfiltered content, which may be harmful if not regulated. Ng (2007) concludes that regardless of intention, if this content goes unfiltered it may “justify”, “encode”, “enact” and “routinize” discrimination amongst targeted groups.

This motivates the creation and popularization of The Don’t Patronize Me! dataset to provide a source for engineers to develop categorization algorithms to progress the protection of targeted groups. The dataset, authored by Perez Almen-dros et al. (2020), is a labelled dataset containing more than 10,000 paragraphs extracted from news stories, country of origin, and keyword which is then labelled indicating its level of Patronizing and Condescending Language (PCL).

I propose an extension to the base-line categorization model ‘RoBERTa’ which beats the initial benchmarks of 0.48 F1-score on the dev-set and 0.49 on the test-set. The repository link is available at the top of this report.

Model	F1 dev-set	F1 val-set
Baseline	0.48	0.49
Mine	0.82	0.57

Table 1: F1 score of PCL language for baseline model compared to final model

## 2 Data Analysis

### 2.1 Feature Distribution

Labels arrive annotated by two main annotators. Their values span from  $[0, 4]$  indicating the level of PCL within the text snippet. Perez Almen-dros et al. (2020) explains how both annotators individually classify each sentence with a score in the range  $[0, 2]$  to indicate the strength of PCL from: no PCL, borderline PCL and blatant PCL. These

scores were then summed to produce a 5-point scale of PCL, where the range  $[0, 1]$  indicates a negative example, and  $[2, 4]$  indicating a positive example.

Data also arrives with labeled country information indicating the source media outlet, and a key-word field. The keyword is provided as input to the model as the search term used to retrieve texts about a target community. A distribution of these fields is shown in Figure 1. Here, the supplied dev dataset contains a disproportionate number of negative (9475) and positive (993) examples of PCL.

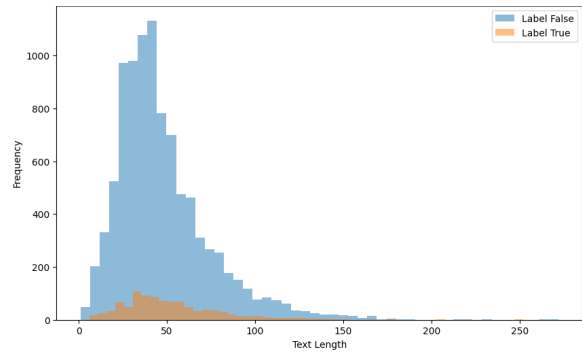


Figure 2: Histogram of word count by PCL (true) and non PCL (false)<sup>1</sup>

Furthermore, the distribution of word counts between the two (See Figure 2) offers no obvious underlying pattern between the two classes. This is because natural language is about how words are said, instead of the quantity.

### 2.2 Subjective Difficulty

Indeed, Perez Almen-dros et al. (2020) mention the subjective nature of the task as it is very subjective. The word ‘lovely’ appears uniquely in PCL text examples. This word isn’t negative, but in the context of the sentence (par\_id: 2404) “We think it’s lovely that so many have come forward to help out a family clearly in need!” the sentence has been labeled as blatantly PCL with a score of 3. In-

<sup>1</sup> Without loss of generality, text with more than 250 words have been clipped

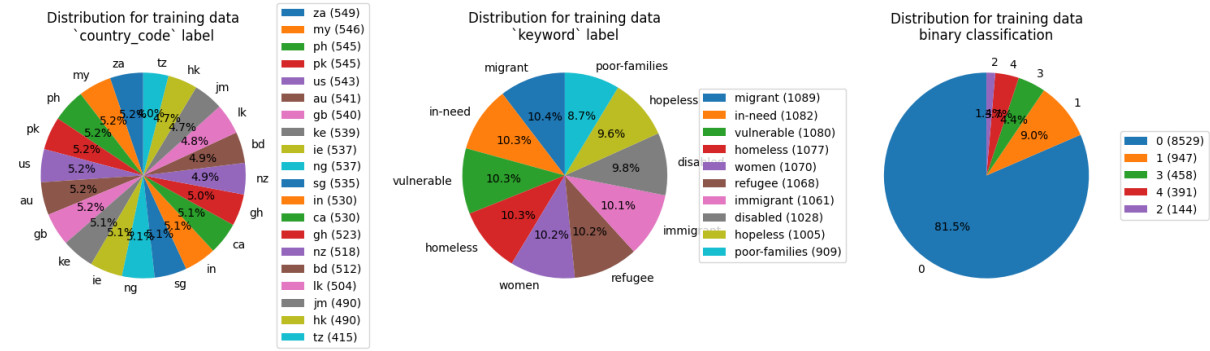


Figure 1: Pie charts of different label distributions in the supplied dev-set

fact, tweaking the sentence from “a family clearly in need” to “families in need” removes entitlement and converts it to compassion.

## 2.3 Artifacts

The flexible environment of news articles allows for authors to post resources and further reading, like in the case of par\_id: 2838: “*Read more about the site’s history here: [LINK]*”. Therefore, with a primitive language processing model, we cannot determine the degree of PCL from a link. Some links and references are also clipped in the source with token “TOOLONG” which also ought to be pre-processed to avoid confusion from the model. Additionally, writers are free to reference other users (par\_id: 5598 “@toekunbore [...]”), the handle of which would likely be ignored by language models.

Lastly, adding to the complexity of processing language, we must also filter and process out-of-vocabulary words. This may appear in the form of a typo like in par\_id: 8221 “*Do they think it is ”” jsutice ”” ti imprison [...]*”, hyperboles like par\_id: 5333 “*But nooooo [...]*”, or slang like in par\_id: 1674 “*Kyle really your a pig, lol [...]*”.

## 3 Minority Upsampling

The data is very imbalanced; Section 2.1 discusses the sparsity of positive PCL examples in the training data. Therefore, it is possible that the model may experience great performance in the majority negative class, and fail to learn details about the minority positive class (Fernández et al. (2018)).

We can synthetically produce more data points for the minority positive class by changing the content of the sample without changing the sentiment of the text.

## 3.1 Failed Considerations

All data augmentations are equal, but some are more equal than others; throughout development, considerations for data augmentations were made, but not all were implemented.

### 3.1.1 Stop Word Removal

Stop words contain little value, so can useful to decrease the length of sentence. However, it may change the sentiment of a sentnece: “I do not like you because I’m better than you” changed into “I like I’m better”, thus removing all PCL sentiment from the sentence, and rendering an ineffective augmentation.

### 3.1.2 Easy Data Augmentation

Wei and Zou (2019) propose a pipeline to tweak sentences by replacing, deleting, swapping and inserting words. The authors comment that by fine-tuning replacement parameters and the up-sampling factor, it can “can boost performance on text classification tasks”. However, we found that the model quickly overfit within one epoch.

### 3.1.3 Rephrasing Models

Further efforts were dedicated to rephrasing the text with the help of a pre-trained GPT-2 model as suggested by Dai et al. (2023). However, significant inconsistencies were discovered depending on the sentence length, so ultimately this effort was abandoned.

## 3.2 Successful Considerations

### 3.2.1 Translation

This upsampling technique is inspired by Nallamuthu (2023) who suggests translating the text into a non-english language and then back into English. The hope is that translating back introduces new wording and structure without augmenting

the sentiment of the sentence. We take this a step further by introducing a parameter to control the depth of translation.

### 3.2.2 Swapping Sentences

Within the same paragraph, we can swap the ordering of sentences. This augmentation was less researched by helped upsample the minority class.

## 4 Data Augmentation

### 4.1 Tokenization

By tokenizing the input, we can efficiently assign [UNKNOWN] tokens to out-of-vocabulary words. This is important for a pre-trained model which are trained on well-formed English text, and may not understand slang or typos.

### 4.2 Masking inputs

Further benefit of tokenization includes the choice of randomly masking a subset of words in the input. This allows models to capture rich contextual information and dependencies in language. This allows the model to generalize well to various text classification tasks, even when faced with different writing styles, vocabulary, and sentence structures.

## 5 Modelling

*For the successful implementation of a classifier model (this could be a transformer or any other ML model of your choice. Do give justification for your choice.):*

### 5.1 Model Choice

**10 marks:** Successful implementation of a model (train and produce predictions which outperform the F1 score for the RoBERTa-base baseline provided). 7 marks for outperforming the baseline model on the official dev set (0.48) and 3 marks for outperforming the baseline model on the test set (0.49).

### 5.2 Hyper-parameters

**5 marks:** Choice of model hyper-parameters and description of your model setup. This should include choosing an appropriate learning rate and checking whether implementing a learning schedule improves performance. Also consider whether your model is cased or uncased. You should mention how many epochs you train the model for, whether you are using any early-stopping, and how you are using the training labels.

## 6 Results

**5 marks:** Description of the model results and your hyper-parameter tuning (some evidence of this is required in your report). Your results should show how the different strategies you have tried impacted the model performance. For any results presented in your paper, you should be clear if these are from your own internal dev set or the official dev set.

### 6.1 Comparison

**10 marks:** Compare your model performance to two simple baselines (e.g. a BoW model). Share some of the features that one of your baseline models used, and highlight an example misclassified with a suggestion of why the baseline may have made the misclassification.

## 7 Analysis

*Analysis questions to be answered (these questions can be answered without training any additional models): Your report should state the analysis questions so that this can be read as a self-contained report, rather than referring to ‘analysis question 1’ etc.*

- 1. 5 marks:** To what extent is the model better at predicting examples with a higher level of patronising content? Justify your answer.
- 2. 5 marks:** How does the length of the input sequence impact the model performance? If there is any difference, speculate why.
- 3. 5 marks:** To what extent does model performance depend on the data categories? E.g. Observations for homeless vs poor-families, etc.

## 8 Conclusion

**5 marks:** Conclusion, with a summary of your results, and your key findings from the analysis questions. You should suggest at least one further experiment as a next step.

## References

- Haixing Dai, Zheng Liu, Wenxiong Liao, Xiaoke Huang, Yihan Cao, Zihao Wu, Lin Zhao, Shaochen Xu, W. Liu, Ninghao Liu, Sheng Li, Dajiang Zhu, Hongmin Cai, Lichao Sun, Quanzheng Li, Dinggang Shen, Tianming Liu, and Xiang Li. 2023. [Auggpt: Leveraging chatgpt for text data augmentation](#).
- Alberto Fernández, Salvador García, Mikel Galar, Ronaldo C. Prati, Bartosz Krawczyk, and Francisco Herrera. 2018. *Learning from Imbalanced Data Sets*. Springer Cham.
- Nandhini Nallamuthu. 2023. [Handling imbalanced data – machine learning, computer vision and nlp](#).
- Sik Hung Ng. 2007. [Language-based discrimination: Blatant and subtle forms](#). *Journal of Language and Social Psychology*, 26(2):106–122.
- 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.
- Jason Wei and Kai Zou. 2019. [EDA: Easy data augmentation techniques for boosting performance on text classification tasks](#). In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 6383–6389, Hong Kong, China. Association for Computational Linguistics.