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

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 Almendros 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
Mine	?	?

Table 1: F1 score of models attempted

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 Almendros 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 keyword field. The keyword is provided as input to the model as the search term used to retrieve texts about a target community.

2.2 Data Distribution

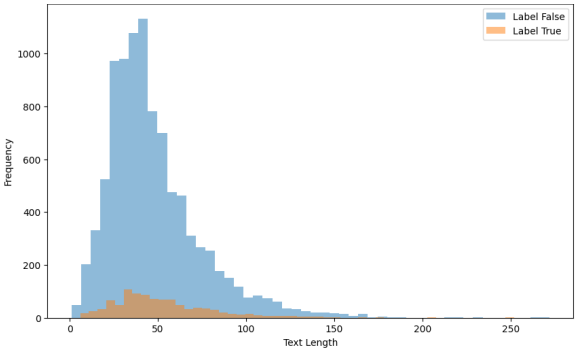

Figure 1: Histogram of word count by PCL (true) and non PCL (false)[1]

The test data contains a disproportionate number of negative (9475) and positive (993) examples of PCL. Furthermore, the distribution of word counts between the two (See Figure 1) offers no obvious underlying pattern between the two classes. This is because natural language is about how words are said, instead of the quantity.

2.3 Subjective Difficulty

Indeed, Perez Almendros 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. Infact, tweaking the sentence from "a family clearly

[1] Without loss of generality, text with more than 250 words have been clipped

in need" to "families in need" removes entitlement and converts it to compassion.

2.4 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 aught 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 Input strategy

The data is very imbalanced; Section 2.1 discusses the sparsity of positive PCL examples in the training data. This less-so because of poor sampling, but rather reflecting the distribution of PCL in media. 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 therefore propose a few methods to increase the amount of training data available to the model (Section 4).

3.1 Pre-processing

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.

3.1.1 Spelling Mistakes

Section 2.4 describes artifacts in the training data that may be problematic in their small quantities. However, we may switch the ordering of letters within words to purposefully mimic grammatical mistakes. Not only will this provide for potentially more data, it also trains the network to be more resilient to spelling mistakes which is an important edge case for models to handle.

3.1.2 Synonym insertion

Another technique for synthetically increasing the number of positive examples is to replace some words with synonyms without changing a sentence's sentiment. For this, we have to have enough synonym variety whilst minimizing the cosine distance between the two swapped words. The hope is to form legitimate sentences with equal PCL scores.

3.1.3 Masking inputs

Finally, an experiment surrounding masking entries in sentences is a potential useful augmentation that may increase the pool of available positive samples. This may help with boundary cases of PCL, as often a few words changed may increase the degree of PCL. Therefore, by masking there is a high chance that irrelevant filler words may be covered which will force more attention on more significant vocabulary.

3.2 Data Sampling

With the above synthetic augmentations to data, we should obtain enough data to evenly batch training with even quantities of negative and positive examples, thus solving the data imbalance issue and allowing for a more balanced F-1 score.

4 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.):

4.1 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).

4.2 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.

5 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.

5.1 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.

6 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.

7 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

Alberto Fernández, Salvador García, Mikel Galar, Ronaldo C. Prati, Bartosz Krawczyk, and Francisco Herrera. 2018. *Learning from Imbalanced Data Sets*. Springer Cham.

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.