Tesla at SemEval-2022 Task 4: Patronizing and Condescending Language Detection using Transformer-based Models with Data Augmentation

Sahil Manoj Bhatt **Manish Shrivastava**
Language Technologies Research Center (LTRC)
International Institute of Information Technology, Hyderabad
sahil.bhatt@research.iiit.ac.in
m.shrivastava@iiit.ac.in

Abstract

This paper describes our system for Task 4 of SemEval 2022: Patronizing and Condescending Language (PCL) Detection. For sub-task 1, where the objective is to classify a text as PCL or non-PCL, we use a T5 Model fine-tuned on the dataset. For sub-task 2, which is a multi-label classification problem, we use a RoBERTa model fine-tuned on the dataset. Given that the key challenge in this task is classification on an imbalanced dataset, our models rely on an augmented dataset that we generate using paraphrasing. We found that these two models yield the best results out of all the other approaches we tried.

1 Introduction

Detecting the presence of patronizing and condescending elements in text is an important task for NLP because of the social impact it has. PCL is characterised by a superior attitude towards others, or a manner of speech that seems to portray others in a pitying way. It is different from other problems in the field of text classification such as detection of hate speech or abusive comments because it is not necessarily done on purpose. Having an automated system that is capable of understanding and classifying language that contains PCL elements would be the first step towards making people and entities, such as media publications, aware of the kind of language they use when talking about vulnerable communities, and as a result, prevent discrimination, stereotypes and harm that could potentially arise from the use of such language.

The PCL Detection task at SemEval-2022 (Pérez-Almendros et al., 2022) aims to solve this problem by exploring different systems that are capable of detecting features that indicate and categorize PCL in the *Don't Patronize Me!* dataset (Pérez-Almendros et al., 2020). Sub-task 1 has been formulated as a binary classification problem, where the goal is to identify whether a given text falls under the category of PCL or not. Sub-task 2 is a multi-label classification problem, where a given text is either free of PCL or belongs to one or more of the seven PCL categories described in the dataset provided.

This paper describes the system developed by team Tesla for SemEval-2022 Task 4. One of the key challenges of this task is the unequal class distribution across the dataset for both sub-tasks.

In this paper, we introduce a system to detect PCL using i) a T5 (Raffel et al., 2019) model for subtask-1 and ii) a RoBERTa (Liu et al., 2019) model for sub-task 2. For our final submission, we use these two models finetuned on an augmented dataset, which we create by generating paraphrases of the sentences belonging to the minority classes in the dataset. Our system ranked 51st out of 78 teams in sub-task 1 and 27th out of 49 teams in sub-task 2, as shown in the leaderboard[1]. Our system for sub-task 2 outperforms the RoBERTa baseline, obtaining an average F1-score of 0.2445 versus 0.1041 for the baseline. All of our code is made publicly available on Github[2].

2 Task Description

The PCL detection task provides participants with the Don't Patronize Me! dataset (Pérez-Almendros et al., 2020), which consists of more than 10,000 paragraphs taken from English-language news stories across 20 different countries. Each of these paragraphs has been annotated to indicate the presence of PCL, and the dataset for sub-task 2 is further annotated with a category label from different classes proposed. These classes are focused on PCL towards vulnerable communities.

The datasets for both the sub-tasks are not balanced. The number of non-PCL examples is nearly 10

[1] https://sites.google.com/view/pcl-detection-semeval2022/ranking

[2] https://github.com/bhattsahil1/pcl_task

Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022), pages 394 - 399
July 14-15, 2022 ©2022 Association for Computational Linguistics

times more than the number of PCL instances in the dataset for sub-task 1. Similarly, the seven classes are not equally represented in the dataset for sub-task 2, and the number of examples that do not belong to any category exceeds those that belong to at least one category by a significant amount.

3 Related Work

Transformer-based approaches such as BERT (Devlin et al., 2019), RoBERTa (Liu et al., 2019), ALBERT (Lan et al., 2019), T5 (Raffel et al., 2019), etc. have shown an impressive performance on a wide range of NLP tasks, including text classification. They have been found to yield good results on text classification that deal with problems such as detection of hate speech (Basile et al., 2019) and offensive language (Zampieri et al., 2019).

The problem of condescending language detection is explored in the TalkDown Dataset (Wang and Potts, 2019), where the authors released an annotated Reddit corpus of condescending linguistic acts in context, along with a BERT model finetuned on the dataset as the baseline.

Data augmentation has been widely used across various machine learning tasks, particularly in those tasks where data is scarce, such as computer vision problems in the medical domain (Sundaram and Hulkund, 2021), (Sandfort et al., 2019). These augmentation approaches allow for a larger and more diverse training dataset.

With respect to text data augmentation, (Bayer et al., 2021) discusses various approaches, both in the data space and feature space. Techniques such as synonym-replacement (Wei and Zou, 2019), embedding replacement (Wang and Yang, 2015), SMOTE (Chawla et al., 2002), generative methods (Yu et al., 2016), (Radford et al., 2018) etc. have been studied for data augmentation.

Many recent works have also used back-translation (Corbeil and Ghadivel, 2020) and paraphrasing as a way to increase training data, such as data in the areas of dialogue-generation (Gao et al., 2020).

4 System Description

4.1 Data

The dataset for sub-task 1 consists of 10469 examples, containing the paragraph ID, article ID, keyword, country, paragraph text, binary PCL label, and original annotator label (on a scale of 0-4). The labels 2,3 and 4 are considered as examples containing PCL, whereas those having labels 0 and 1 are negative (non-PCL) examples. The distribution can be seen in Table 1. The dataset in sub-task 2 follows a similar format and consists of 9368 training examples. Here the label is a one-hot encoding of the seven classes to which the text may or may not belong. These classes are: *unbalanced power relations*, *shallow solution*, *presupposition*, *authority voice*, *metaphors*, *compassion* and *the poorer the merrier*. The major challenge that both these datasets pose is the lack of positive samples. The dataset, such as the one seen in Table 1, has a positive to negative class ratio of nearly 1:10. This presents difficulties in learning features that characterize PCL since many of the existing techniques don't perform well when there is class imbalance (Madabushi et al., 2020), which is noticeable if the training and test data are dissimilar.

Class	**Instances**
PCL	993
Non-PCL	9476

Table 1: Class distribution for sub-task 1 dataset

Class	**Instances**
Unbalanced power relations (unb)	1290
Shallow solution (sha)	356
Presupposition (pre)	386
Authority voice (aut)	422
Metaphors (met)	342
Compassion (com)	832
The poorer the merrier (the)	69
(None of the seven classes)	7581

Table 2: Class distribution for sub-task 2 dataset

4.2 Data Augmentation

One of the most common problems that are faced in classification problems is the lack of data across different classes. This is an even bigger problem in the case of text classification problems, since generating new samples is not a trivial task.

Undersampling the majority class would lead to a loss of negative samples and under-utilization of the given data, hence the approach we take is on the lines of oversampling minority classes to augment data. However, instead of directly oversampling minority samples, we use a T5 model[3] finetuned for the task of paraphrase generation on the PAWS

[3] https://huggingface.co/Vamsi/T5_Paraphrase_Paws

Class	Instances
PCL	8758
Non-PCL	9476

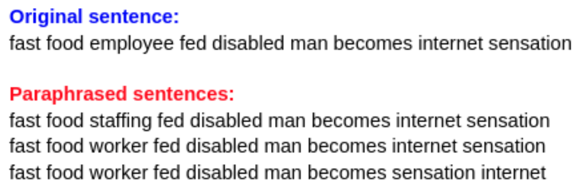
Table 3: Class distribution for sub-task 1 dataset after augmentation

Class	Instances
Unbalanced power relations (unb)	5114
Shallow solution (sha)	1413
Presupposition (pre)	1532
Authority voice (aut)	1682
Metaphors (met)	1361
Compassion (com)	3307
The poorer the merrier (the)	274
(None of the seven classes)	7581

Table 4: Class distribution for sub-task 2 dataset after augmentation

dataset (Zhang et al., 2019). The idea is to generate samples that are similar to the original text, but not the same, as we would like to avoid overfitting that could result from simple oversampling.

We use top-k sampling, in combination with top-p sampling, setting the values of k=120 and p=0.95. For sub-task 1, we generate a maximum of 8 samples for each of the paragraphs belonging to the PCL class. For sub-task 2, we generate a maximum of 3 samples for every sentence that belongs to at least one of the seven class labels. We finally get 18234 examples for sub-task 1 and 14667 examples for sub-task 2.



Original sentence:
fast food employee fed disabled man becomes internet sensation

Paraphrased sentences:
fast food staffing fed disabled man becomes internet sensation
fast food worker fed disabled man becomes internet sensation
fast food worker fed disabled man becomes sensation internet

Figure 1: An example of the paraphrases generated for one of the sentences in the dataset (after pre-processing).

4.3 Pre-processing

We remove punctuation and numbers from our text, and convert each of our sentences to lowercase before using them to generate paraphrases. We remove stop-words from the sentences after carrying out data augmentation.

5 Models

We fine-tune a range of pretrained models, described below, given their good performance on a wide range of NLP tasks. We use the transformer model implementations provided by `simpletransformers`[4] with the default hyperparameters.

BERT (Devlin et al., 2019): We try out the BERT$_{\text{BASE}}$ (uncased) model, which consists of 12 transformer layers, 12 self-attention heads per layer, and a hidden size of 768.

RoBERTa (Liu et al., 2019): We try out the RoBERTa$_{\text{BASE}}$ model. Similar to BERT$_{\text{BASE}}$, RoBERTa$_{\text{BASE}}$ also consists of 12 transformer layers, 12 self-attention heads per layer, and a hidden size of 768.

T5 (Raffel et al., 2019): We use the Text-to-Text Transfer Transformer (T5) released by the authors. We use the T5$_{\text{BASE}}$ model, which has about 220 million parameters, nearly twice the number of parameters in BERT$_{\text{BASE}}$.

We test all three models for sub-task 1, and for sub-task 2 we only try out BERT$_{\text{BASE}}$ and RoBERTa$_{\text{BASE}}$.

6 Experiments

6.1 Implementation details

For both the sub-tasks, we concatenate the paragraph text and the keyword associated with each sample, to explore if certain keywords have an effect on the sample being classified as PCL or not.

We train each of the models with two different conditions - using a dataset without paraphrased instances (original dataset) and a dataset with the original and the paraphrased sentences (augmented dataset). We train them for a single epoch only, to avoid overfitting since the data available is less.

We use an 80:20 train-dev split, preserving the class ratio. We use this to evaluate our model performance in sub-task 1. For sub-task 2 too, we use an 80:20 split for training and evaluating the model. For our final submissions, we train the models for both sub-task 1 and sub-task 2 on the entire dataset.

6.2 Metrics

We report the F1-score, Precision and Recall for sub-task 1. For sub-task 2, we report the F1-score across each of the classes, along with the average score.

[4] `https://simpletransformers.ai/`

7 Results

Before submitting the final models, we evaluated each of the models' performances on validation sets.

Table 5 presents a comparison between all the different models tried out in sub-task 1. The T5$_{\text{BASE}}$ model yields the best F1-score among models trained on the original dataset, and the model also seems to perform decently well on the augmented dataset. The results presented in the table for models trained on the augmented dataset for both the sub-tasks are high since the validation set also contains paraphrases of sentences it might have already seen in the training set. Nevertheless, the relative scores between models trained on the augmented dataset are still a good indicator of their expected performance.

Table 6 discusses the results obtained using BERT$_{\text{BASE}}$ and RoBERTa$_{\text{BASE}}$. The average F1-score reported on the validation set does not change significantly across the four models used, however, the F1-scores for individual classes are significantly different when comparing the models trained on the original dataset versus models trained on the augmented one.

We present the results of our final submissions in Table 7 and 8.

8 Discussion and Conclusion

The T5$_{\text{BASE}}$ models that we submitted for sub-task 1 do not perform well on the test set. One reason for this could be that the T5$_{\text{BASE}}$ should have been trained for longer than one epoch. Another reason could be that BERT-based approaches (BERT, RoBERTa) might be better suited for this task. In addition to this, the results of T5$_{\text{BASE}}$ fine-tuned on the augmented dataset are not very good either, which could be due to overfitting that results from seeing many paraphrased instances of the same sentence during training. A manual inspection of the generated paraphrases also reveals that their quality needs improvement, since many of the generated paraphrases do not differ much from each other, and at times the generation gets reduced to simple oversampling.

The results of sub-task 2 are encouraging and we can see a significant improvement in F1-scores across almost all classes when we consider the RoBERTa$_{\text{BASE}}$ model trained on the augmented dataset. This confirms that augmenting the dataset through paraphrasing does have a positive effect on model performance. Ensuring dissimilarity in the generated paraphrases, choosing an ideal number of paraphrases to generate, and using other techniques to handle imbalanced data such as cost-sensitive learning or augmentation through other methods (or a combination of them), might yield better results.

We thus, see that identifying PCL and categorizing its occurrences is feasible despite its subjective nature, and that transformer-based approaches are capable of doing this.

Acknowledgments

We thank the organisers of this task for their effort, and the anonymous reviewers for their feedback and insightful comments.

References

Valerio Basile, Cristina Bosco, Elisabetta Fersini, Debora Nozza, Viviana Patti, Francisco Manuel Rangel Pardo, Paolo Rosso, and Manuela Sanguinetti. 2019. SemEval-2019 task 5: Multilingual detection of hate speech against immigrants and women in Twitter. In *Proceedings of the 13th International Workshop on Semantic Evaluation*, pages 54–63, Minneapolis, Minnesota, USA. Association for Computational Linguistics.

Markus Bayer, Marc-André Kaufhold, and Christian Reuter. 2021. A survey on data augmentation for text classification.

N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer. 2002. Smote: Synthetic minority oversampling technique. *Journal of Artificial Intelligence Research*, 16:321–357.

Jean-Philippe Corbeil and Hadi Abdi Ghadivel. 2020. BET: A backtranslation approach for easy data augmentation in transformer-based paraphrase identification context. *CoRR*, abs/2009.12452.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Silin Gao, Yichi Zhang, Zhijian Ou, and Zhou Yu. 2020. Paraphrase augmented task-oriented dialog generation.

Model (Sub-task 1)	Metrics (validation set results)		
	Recall	Precision	F1-score
BERT base (original dataset)	0.578	0.789	0.609
RoBERTa base (original dataset)	0.500	0.452	0.475
T5 base (original dataset)	0.694	0.703	0.699
BERT base (augmented dataset)	0.939	0.938	0.938
RoBERTa base (augmented dataset)	0.878	0.879	0.877
T5 base (augmented dataset)	0.870	0.879	0.866

Table 5: Sub-task 1: The results mentioned here are for the validation set from an 80:20 training-validation split of the dataset (with and without augmentation). Do note that the validation set for the original dataset and augmented dataset is different, which is why we are only looking at the relative performance of models on the augmented dataset and not comparing it with models trained on the original dataset.

Model (Sub-task 2)	F1 scores (validation set results)							
	unb	sha	pre	aut	met	com	the	avg
BERT base (original dataset)	0.404	0	0	0	0	0.118	0	0.075
RoBERTa base (original dataset)	0.485	0	0	0	0	0.105	0	0.084
BERT base (augmented dataset)	0.196	0.025	0.079	0.044	0.027	0.116	0	0.070
RoBERTa base (augmented dataset)	0.187	0.039	0.081	0	0.031	0.135	0	0.067

Table 6: Sub-task 2 : The results mentioned here are for the validation set from an 80:20 training-validation split of the dataset (with and without augmentation). The classes mentioned here are abbreviations of classes discussed in Table 2. Do note that the validation set for the original dataset and augmented dataset is different, which is why we are only looking at the relative performance of models on the augmented dataset and not comparing it with models trained on the original dataset.

Model (Sub-task 1)	Metrics		
	Recall	Precision	F1-score
RoBERTa baseline	0.653	0.3935	0.4911
T5 base (original dataset)	0.691	0.283	0.402
T5 base (augmented dataset)	0.577	0.359	0.443

Table 7: Final results for sub-task 1

Model (Sub-task 2)	F1 scores							
	unb	sha	pre	aut	met	com	the	avg
RoBERTa baseline	0.3535	0	0.1667	0	0	0.2087	0	0.1041
RoBERTa base (original dataset)	0.286	0	0	0	0	0	0	0.041
RoBERTa base (augmented dataset)	0.437	0.383	0.163	0.192	0.179	0.357	0	0.2445

Table 8: Final results for sub-task 2

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2019. ALBERT: A lite BERT for self-supervised learning of language representations. *CoRR*, abs/1909.11942.

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.

Harish Tayyar Madabushi, Elena Kochkina, and Michael Castelle. 2020. Cost-sensitive BERT for generalisable sentence classification with imbalanced data. *CoRR*, abs/2003.11563.

Carla Pérez-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.

Carla Pérez-Almendros, Luis Espinosa-Anke, and Steven Schockaert. 2022. SemEval-2022 Task 4: Patronizing and Condescending Language Detection. In *Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022)*. Association for Computational Linguistics.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2018. Language models are unsupervised multitask learners.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2019. Exploring the limits of transfer learning with a unified text-to-text transformer. *CoRR*, abs/1910.10683.

Veit Sandfort, Ke Yan, Perry J. Pickhardt, and Ronald M. Summers. 2019. Data augmentation using generative adversarial networks (cyclegan) to improve generalizability in ct segmentation tasks.

Shobhita Sundaram and Neha Hulkund. 2021. Gan-based data augmentation for chest x-ray classification.

William Yang Wang and Diyi Yang. 2015. That's so annoying!!!: A lexical and frame-semantic embedding based data augmentation approach to automatic categorization of annoying behaviors using #petpeeve tweets. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 2557–2563, Lisbon, Portugal. Association for Computational Linguistics.

Zijian Wang and Christopher Potts. 2019. TalkDown: A corpus for condescension detection in context. 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 3711–3719, Hong Kong, China. Association for 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 6382–6388, Hong Kong, China. Association for Computational Linguistics.

Lantao Yu, Weinan Zhang, Jun Wang, and Yong Yu. 2016. Seqgan: Sequence generative adversarial nets with policy gradient. *CoRR*, abs/1609.05473.

Marcos Zampieri, Shervin Malmasi, Preslav Nakov, Sara Rosenthal, Noura Farra, and Ritesh Kumar. 2019. SemEval-2019 task 6: Identifying and categorizing offensive language in social media (OffensEval). In *Proceedings of the 13th International Workshop on Semantic Evaluation*, pages 75–86, Minneapolis, Minnesota, USA. Association for Computational Linguistics.

Yuan Zhang, Jason Baldridge, and Luheng He. 2019. Paws: Paraphrase adversaries from word scrambling.