AliEdalat at SemEval-2022 Task 4: Patronizing and Condescending Language Detection using Fine-tuned Language Models, BERT+BiGRU, and Ensemble Models

Ali Edalat, Yadollah Yaghoobzadeh, Behnam Bahrak
School of Electrical and Computer Engineering, College of Engineering
University of Tehran, Tehran, Iran
{ali.edalat,y.yaghoobzadeh,bahrak}@ut.ac.ir

Abstract

This paper presents the AliEdalat team's methodology and results in SemEval-2022 Task 4: Patronizing and Condescending Language (PCL) Detection. This task aims to detect the presence of PCL and PCL categories in text in order to prevent further discrimination against vulnerable communities. We use an ensemble of three basic models to detect the presence of PCL: fine-tuned bigbird, fine-tuned mpnet, and BERT+BiGRU. The ensemble model performs worse than the baseline due to overfitting and achieves an F1-score of 0.3031. We offer another solution to resolve the submitted model's problem. We consider the different categories of PCL separately. To detect each category of PCL, we act like a PCL detector. Instead of BERT+BiGRU, we use fine-tuned roberta in the models. In PCL category detection, our model outperforms the baseline model and achieves an F1-score of 0.2531. We also present new models for detecting two categories of PCL that outperform the submitted models.

1 Introduction

Increasing internet access rates and the development of a diverse range of online forums have allowed people around the world to engage in a tremendous range of topics. This has been accompanied by an increase in unhealthy online texts whose negative effects on people have been significant. One type of such unhealthy texts is a text with patronizing and condescending language (PCL). When a person's language expresses a superior attitude towards others or describes their situation in a benevolent way that creates a sense of pity, the person has used this type of language. In the media, vulnerable communities seem to be a good target for this type of language. However, this type of language can normalize discrimination. We believe that unfair treatment of vulnerable groups leads to greater deprivation and inequality for these groups. Therefore, recognizing the existence of this type of language and its variations is important and can prevent these problems.

So far, significant work has been done on modeling the language that deliberately and openly undermines others, such as offensive language or hate speech, but little has been done on the language of humiliation and pity. This language of humiliation and pity is used in the media subtly and indirectly and different from other types of unhealthy languages. The special focus on the language of humiliation and compassion for vulnerable communities has been noted only in the work of Pérez-Almendros et al., (2020). In this work, a dataset to identify this type of language is presented, but no significant work has been done in designing a model to classify this type of text.

Unhealthy text papers usually focus on obvious and aggressive phenomena such as detecting fake news, fact-checking, modeling offensive language, and spreading rumors. There has been a few work on PCL recently. Wang and Potts (2019) introduced compassion modeling in direct communication from the perspective of natural language analysis. They created and tagged a dataset with social media messages. Sap et al. (2019) Discussed the specific uses of language and power, especially the unbalanced power relations often present in degrading treatment, and the social consequences of these applications. Unfair treatment of disadvantaged groups was also examined as an example of these cases. Price et al. (2020) Provided datasets for classifying unhealthy speech on social media. They provided fine-grained classifications for all kinds of unhealthy writings, one of which was PCL.

Of course, the use of this type of writing is not limited to weak groups in society. There is still a need to design a model to detect such language towards vulnerable communities. PCL is a toxic

Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022), pages 387 - 393

language that implicitly has a negative impact on public opinion. There are tasks that generally identify toxic language that can be used to provide an answer to this problem. For example Lees et al. (2021) proposed the use of a fine-tuned BERT model to detect veiled toxicity.

To design a model to detect this language in vulnerable communities, we participated in SemEval 2022 task 4 competition (Pérez-Almendros et al., 2022). This paper describes the models we provide for detecting PCL. The contest data is taken from the work of Pérez-Almendros et al., (2020).

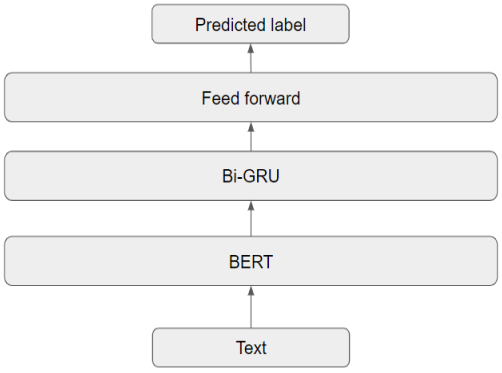

Figure 1 The general structure of BERT+BiGRU model

To detect the presence of PCL, we present an ensemble model. This model consists of three basic models: fine-tuned bigbird, fine-tuned mpnet and BERT + BiGRU. The models are combined with the weighted average. This model cannot perform better than the baseline because our model is overfitted. In this paper, we present a solution to this model's problem and create a new model that can achieve an F1-score of 0.5505. To identify existing PCL categories, we consider the categories separately. To detect the presence of any category, we act like detecting the presence of PCL. The basic models for making ensemble models are fine-tuned bigbird, fine-tuned mpnet and fine-tuned roberta. The model can outperform the baseline model. In addition to this model, in two categories, we improve the diagnostic model and build a new model. This model can achieve an F1-score of 0.3160. The statement of contributions is given below.

To balance the data set, we used a different method than the data set providers. Instead of using the sampling method (We sample twice the number of PCL data from non-PCL data), we used a combination of the sampling method and EDA (Wei et al., 2019). And in compassion and metaphor diagnosis, we used a set of related articles for balancing. We paid attention to the medium and high lengths of the texts and used language models with the ability to summarize long texts. We used the ensemble model for classification to help reduce the bias caused by the data set imbalance.

2 Models

In this section, we describe how to detect the presence of PCL in the text and how to detect the type of PCL in the text. We describe the models used for these diagnoses.

2.1 Subtask1: PCL detection

Recognizing the presence of PCL in the text is a two-class classification problem. To do this, we use a model that is an ensemble of three basic models. Our base models are fine-tuned bigbird, fine-tuned mpnet, and BERT+BiGRU model. We use BERT language model (Devlin et al., 2018) for this classification to prepare the BERT+BiGRU model. We also fine-tune the Big Bird language model (Zaheer et al., 2020) and the MPNet language model (Song et al., 2020) to prepare the fine-tuned bigbird and fine-tuned mpnet models.

The texts are taken from the news. For this reason, there is medium to long texts in the data, and to address this issue, we used two language models, mpnet and bigbird, to create our model. This allows the model for long texts to extract the information needed for classification. Details about the length of the texts are given in the section 4.1 .

Each of the basic models learn separately on the training data. We then combine the results of the models to predict the text class using the weighted average to generate our model prediction. To combine the models, we use the probability that the text has PCL. Each model predicts this probability for each text. First, we use the weighted average of the probabilities predicted by these two models. We use a weight of 0.4 for the mpnet model prediction and a weight of 0.6 for the bigbird model prediction. Then, we use the weighted average to combine the BERT+BiGRU model prediction and the average prediction of the previous two models. The weight of the BERT+BiGRU model in this average is equal to 0.3 and the weight of the combination of the two previous models is equal to 0.7.

2.2 BERT+BiGRU model

The model consists of three layers. The first layer of the model applies BERT. In this layer, we give the cleaned text to the language model and get the embedding of text tokens. Then we give tokens' embedding to the Bi-GRU layer. The output of the Bi-GRU layer is then given to the feed forward layer to predict the input class. The general structure of this model is shown in Figure 1. To clear the text, we remove the HTML tags, URLs, Mentions and Emojis in the text. More details of this model are given in Section 3.1 .

2.3 Fine-tuned other Language Models

To fine-tune these language models, we use a two-layer model. In the first layer, the language model takes the input text and creates a display for the entire text. The classifier token embedding is used to display the entire text. In the second layer, we predict the label using a feed forward network. In these models we do not clean the input text. Figure 2 shows the general structure of the model to fine-tune the language model.

2.4 Subtask2: PCL categories detection

The PCL categories detection problem is a multi-label classification. Given a paragraph, a system must identify which PCL categories (if any) appear in the paragraph. The problem is, a text can have multiple categories at the same time.

To solve this problem, we detect the presence of each category in the text separately from the other categories. That is, we create a separate model to identify each category. Each model solves a binary classification problem. This model determines whether the text has the desired PCL category or not.

To identify the "Unbalanced Power Relations" category in the text, we use an ensemble of two basic models. We use fine-tuned bigbird and fine-tuned mpnet as basic models. We use a weighted average to combine the two models. On this ensemble, the bigbird model weighs 0.7 and the other model weighs 0.3.

To identify the "Shallow Solution" category in the text, we also use an ensemble of two basic models. We use fine-tuned roberta and fine-tuned mpnet as basic models. We use a weighted average to combine the two models. On this ensemble, the roberta model weighs 0.7 and the other model weighs 0.3. We fine-tune the RoBERTa language model (Liu et al., 2019) for this classification to prepare the fine-tuned roberta model.

To identify the "Presupposition" category in the text, we use an ensemble of two basic models. We use fine-tuned bigbird and fine-tuned mpnet as basic models. We use a weighted average to combine the two models. On this ensemble, the bigbird model weighs 0.7 and the other model weighs 0.3 and the sum of the weights is one. In bigbird for this category, the error weight for class with "Presupposition" is 4 times that of class without "Presupposition".

To identify the "Authority Voice" and "Metaphor" categories in the text, the model structure is similar to the "Presupposition" detection model in the text. The only difference between the detection models of these categories is in the weights of the base models to create the ensemble model. In the weighted average for the "Authority Voice" category, the weights of the bigbird model and the mpnet model are 0.5 and 0.5. For the "Metaphor" category, the weight of these models are 0.6 and 0.4, respectively.

To identify the "Compassion" category in the text, we use an ensemble of three basic models. First, we combine the results of the two basic models with the weighted average. These basic models are fine-tuned bigbird and fine-tuned mpnet. We use a weight of 0.4 for the bigbird model and a weight of 0.6 for the other model. Then we combine the result of combining the previous two models with the prediction of the fine-tuned roberta model. We use a weighted average with a weight of 0.1 for the roberta model and we set the weight of the combination of the previous two models to 0.9.

We also use fine-tuned roberta to identify the "The Poorer The Merrier" category in the text.

Task 1 and Task 2 share the same input paragraphs and have different labels respectively. The reason we chose Task 1 fine-tuning models is the same as the reason for using Task 2 models. In addition to the Task 1 models, we also used the RoBERTa model for Task 2, which is the base model presented in the competition. In each category, all of these models are trained for classification, and we presented the best possible combination of these models as the final model. To determine the weights for creating the ensemble model, the performance of the constituent models has been considered. The model with better

performance has more weight in the ensemble model.

3 Experimental Setup[1]

In this section, the structural details of the base models are given. All models are trained on the GPU of google colab[2] in normal account mode.

3.1 BERT+BiGRU model

In the Bi-GRU layer of this model, we use two layers. Set the dimension of the hidden layer vector to 256. The direction of one GRU (Chung et al., 2014) is the positive direction of the input sequence (from left to right), and the other is the reverse direction of the input sequence (from right to left). When feature extraction is performed on the input sequence, the GRUs in the two directions do not share the state. The state transition rules of GRU follow the transition occurrence between the same states. However, at the same moment, the output results of the GRUs in the two directions are spliced as the output of the entire Bi-GRU layer. We apply dropout to the output of this layer with a probability of 0.25.

We output the Bi-GRU layer result to the feed forward network. This feed forward network consists of a hidden layer with 256 neurons. We also set the maximum number of input tokens to 512. In the learning process of this model, we use 5 epochs. In learning phase, the error weight for PCL class is 2 times that of non-PCL class.

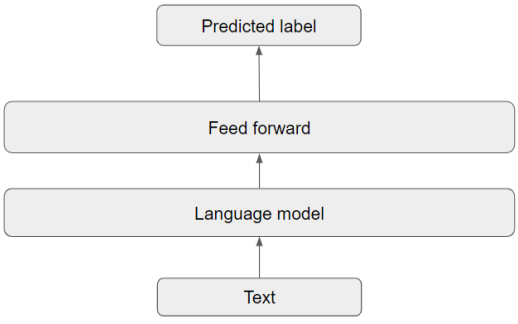

Figure 2 general structure of our model for fine-tuning the language model

3.2 Fine-tuned other Language Models

In the learning process of this model, we use 1 epoch. To fine-tune these language models, we use the ClassificationModel in the simpletransformers[3] library. The weight of the error in predicting the sample of class with label 1 can be different from the class with label 0.

4 Results and Analysis

4.1 Dataset

We use the SemEval 2022 task 4 dataset. We have three sets of training, evaluation, and testing in the competition data. The training dataset is imbalanced for both sub-tasks. Task 1 and Task 2 share the same input paragraphs and have different labels respectively. The maximum, mean, and median length of training texts are 5518, 294, and 258. Length means the number of characters. The maximum, mean, and median number of words in training texts are 911, 53, and 45.

To solve the problem of class imbalance in the learning process, we use the augmentation methods provided for toxic texts (Juuti et al., 2020). Among these methods, we use the EDA method for all binary classification problems. In some cases, we use other relevant datasets to increase minority class data.

In Task 1, we consider a constant difference of 4900 sample between the data number of the two classes. For classification, we paste the paragraph text, the keyword corresponding to the text, and the full name of the country associated with it, and consider it as the text for the classification. To reduce the data difference between the two classes, which is more than 4900, we add the texts of the Task 2 dataset that are not in the Task 1 dataset. We also add the first 100 texts of the talkdown dataset (Wang and Potts, 2019) to the collection. Wang and Potts (2019) introduced compassion modeling in direct communication from the perspective of natural language analysis. They created and tagged talkdown dataset with social media messages. Compassion is a type of PCL. For this reason, the use of compassion data helps detect the presence of PCL. All texts with a PCL type are added. We do not add the text itself, but we use the modified text

[1] Our code can be found at: https://github.com/AliEdalat/SemEval-2022-task-4-PCL-detection.git

[2] https://colab.research.google.com/

[3] https://simpletransformers.ai/docs/classification-models/

by substituting several words with the same meaning by using WordNet (Miller, 1995). We fill the rest of the difference between the two classes with two EDA methods. One way is to use modified texts that have PCL, by replacing some words with their synonyms in WordNet. Another way is to use modified texts that have PCL, by replacing some words with their nearest neighbours in Glove (Pennington et al., 2014) embedding space. Glove is a pre-trained word embedding that is trained on Twitter data.

For each category in Task 2, except for the "Metaphor" and "Compassion" categories, as in Task 1, we balance the classes. There are only two differences. We consider the difference between the two classes to be 5900 and do not use any other datasets to generate data. For the "Metaphor" category, the difference between the two categories is 5900. We get help from 1200 datapoints from vumc (Mu et al., 2019) dataset for balancing. Like the first task, we do not use this data itself and modify the text using WordNet. We fill in the rest of the gap like the other categories. For the "Compassion" category, we act like the "Metaphor" category. The only difference is the use of talkdown dataset instead of vumc.

With these methods, we prepare training data. To fine-tune non-BERT language models, we sample twice the amount of class with label 1 data from class with label 0. To predict the test data, we add the data that corresponds to the problem, with label 1 from the evaluation data to the training data.

4.2 Evaluation Metrics

We use competition metrics to evaluate system. For each binary classification, we use F1 score for label 1 as the evaluation metric. For Task 2, we use the mean F1 score for all categories as the evaluation metric.

4.3 Results

The results of the proposed model for Task 1 are given in the Table 2. In addition to the results of our model, the results of the baseline model of the competition are also included. The baseline model is fine-tuned roberta. This model uses sampling to balance the training dataset. In addition to our model, our model without using BERT+BiGRU in creating ensemble model is included ("(BigBird, MPNet)" shows this model in Table 2).

As you can see, our model did not perform well in the test and performed worse than the baseline due to overfit. But our model outperformed the baseline in evaluation. The reason for this overfit was the addition of the BERT+BiGRU model. As can be seen, if we remove BERT+BiGRU from the model, the model performs better in testing and evaluation than the baseline. BERT+BiGRU has contributed a little to the performance of the model in evaluation, but it has overfitted our model. So, the best model to solve this problem is "(BigBird, MPNet)". Using this model, we achieved an F1-score of 0.5505.

Model	Eval F1	Test F1
(BigBird, MPNet, BERT+BiGRU) submitted system	0.5965	0.3031
(BigBird, MPNet)	0.5789	0.5505
Baseline	0.4829	0.4911

Table 2 The results of the proposed model for Task 1.

Category	Model	Eval f1	Test f1
Metaphor	**Our, submitted system (MPNet, BigBird 1:4)**	0.4557	0.1345
	PCM (MPNet, BigBird)	-	0.2947
Compassion	**Our, submitted system (BigBird, MPNet, RoBERTa)**	0.5565	0.1129
	PCM (BigBird, MPNet)	-	0.3932

Table 1 results of our models in the "Compassion" and "Metaphor" categories.

The performance of our model for Task 2 in evaluation and testing is given in the Table 3. The model presented in the Models section is called "Our Model" is listed in Table 3. The "Problem-free model in Compassion and Metaphor (PCM)" model is similar to our model, except that for the

"Compassion" category, we remove the fine-tuned roberta base model from its detection model. For the "Metaphor" category, we also set the weight of the class 1 detection error equal to the weight of the other class. As can be seen, our two models performed better than the baseline. The "PCM" model performs better than the "Our" model. Unfortunately, we did not use the "PCM" model in this competition. The reason for not using more models in two tasks was the restriction on uploading answers in the contest. Our F1-score in Task 2 of this competition was 0.2531. If we used the "PCM" model, our F1-score would be 0.3160.

The results of our two models in the

Model	Eval f1	Test f1
Our Model (submitted system)	0.3677	0.2531
Problem-free model in Compassion and Metaphor (PCM)	-	0.3160
Baseline	0.1340	0.1041

Table 3 The performance of our model for Task 2.

"Compassion" and "Metaphor" categories are shown in the Table 1. BigBird 1:4 means that the two classes zero and one weigh one and four in the fine-tuned bigbird, respectively.

As can be seen, the performance of "Our Model" for the two classes in the test phase was very different from our performance in the evaluation phase. The performance of the "PCM" model in the test phase was much better than the Our Model. The difference in the performance of "Our Model" in the two stages of evaluation and testing in the "Compassion" category was due to the use of the fine-tuned model roberta as the base model. Using this model has caused overfit. The reason for the difference in performance in the "Metaphor" category was due to the different weight of the class with metaphor error compared to the class without metaphor in the bigbird model. This different weight has created a bias for our model as a whole. As can be seen, by solving these problems, the "PCM" model was able to perform better than "Our Model"

5 Conclusion

In this paper, we presented models for two tasks. For Task 1, we presented an ensemble model consisting of three basic models. We reviewed the results of this model in the competition. We examined the weaknesses of this model and presented another model with a similar structure that performed better on the test data. For Task 2, we considered identifying each category separately from the other categories. We provided a model to identify each category. We examined the result of our prediction based on these models in the competition and identified weaknesses. By solving these cases, we changed the classification model of the two categories. We were able to come up with a new prediction for test data that would have a better result than our original model. Using our second model resulted in better ranks in the testing phase. In future work for the second task, the categories can be considered related. This is because some categories have a common concept. The extracted features according to other categories, can be used to classify a category. In future work, other ways can be proposed to solve the problem of class imbalance. For example, constructor models can be used to create text for a class on a conditional basis.

References

Devlin, J., Chang, M.W., Lee, K. and Toutanova, K., 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805.

Zaheer, M., Guruganesh, G., Dubey, K.A., Ainslie, J., Alberti, C., Ontanon, S., Pham, P., Ravula, A., Wang, Q., Yang, L. and Ahmed, A., 2020. Big bird: Transformers for longer sequences. Advances in Neural Information Processing Systems, 33, pp.17283-17297.

Song, K., Tan, X., Qin, T., Lu, J. and Liu, T.Y., 2020. Mpnet: Masked and permuted pre-training for language understanding. Advances in Neural Information Processing Systems, 33, pp.16857-16867.

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L. and Stoyanov, V., 2019. Roberta: A robustly optimized bert pretraining approach. arXiv preprint arXiv:1907.11692.

Chung, J., Gulcehre, C., Cho, K. and Bengio, Y., 2014. Empirical evaluation of gated recurrent neural

networks on sequence modeling. arXiv preprint arXiv:1412.3555.

Juuti, M., Gröndahl, T., Flanagan, A. and Asokan, N., 2020. A little goes a long way: Improving toxic language classification despite data scarcity. Findings of the Association for Computational Linguistics: EMNLP 2020,.

Wei, J. and Zou, K., 2019. EDA: Easy Data Augmentation Techniques for Boosting Performance on Text Classification Tasks. Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP),.

Mu, J., Yannakoudakis, H. en Shutova, E. (2019) "Learning outside the box: Discourse-level features improve metaphor identification", in Proceedings of the 2019 Conference of the North. Proceedings of the 2019 Conference of the North, Stroudsburg, PA, USA: Associa-tion for Computational Linguistics. doi: 10.18653/v1/n19-1059.

Pérez-Almendros, C., Espinosa-Anke, L. en Schockaert, S. (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, bll 5891–5902.

Wang, Z. and Potts, C., 2019. TalkDown: A Corpus for Condescension Detection in Context. Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP),.

Sap, M., Gabriel, S., Qin, L., Jurafsky, D., Smith, N.A. and Choi, Y., 2019. Social bias frames: Reasoning about social and power implications of language. arXiv preprint arXiv:1911.03891.

Price, I., Gifford-Moore, J., Flemming, J., Musker, S., Roichman, M., Sylvain, G., Thain, N., Dixon, L. and Sorensen, J., 2020. Six Attributes of Unhealthy Conversations. Proceedings of the Fourth Workshop on Online Abuse and Harms,.

Pérez-Almendros, C., Espinosa-Anke, L. en Schockaert, S. (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.

Lees, A., Borkan, D., Kivlichan, I., Nario, J. and Goyal, T., 2021, April. Capturing Covertly Toxic Speech via Crowdsourcing. In Proceedings of the First Workshop on Bridging Human–Computer Interaction and Natural Language Processing (pp. 14-20).

Miller, G.A., 1995. WordNet: a lexical database for English. Communications of the ACM, 38(11), pp.39-41.

Pennington, J., Socher, R. and Manning, C.D., 2014, October. Glove: Global vectors for word representation. In Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP) (pp. 1532-1543).