HITMI&T at SemEval-2022 Task 4: Investigating Task-Adaptive Pretraining And Attention Mechanism On PCL Detection

Zihang Liu,Yancheng He,Feiqing Zhuang,Bing Xu*
Machine Intelligence and Translation Laboratory,
Harbin Institute of Technology, Harbin, China
21s103280@stu.hit.edu.cn, 21s103127@stu.hit.edu.cn,
21s003015@stu.hit.edu.cn, hitxb@hit.edu.cn

Abstract

This paper describes the system for the Semeval-2022 Task4 "Patronizing and Condescending Language Detection".An entity engages in Patronizing and Condescending Language(PCL) when its language use shows a superior attitude towards others or depicts them in a compassionate way. The task contains two parts. The first one is to identify whether the sentence is PCL, and the second one is to categorize PCL. Through experimental verification, the RoBERTa-based model will be used in our system. Respectively, for subtask 1, that is, to judge whether a sentence is PCL, the method of retraining the model with specific task data is adopted, and the method of splicing [CLS] and the keyword representation of the last three layers as the representation of the sentence; for subtask 2, that is, to judge the PCL type of the sentence, in addition to using the same method as task1, the method of selecting a special loss for Multi-label text classification is applied. We give a clear ablation experiment and give the effect of each method on the final result. Our project ranked 11th out of 79 teams participating in subtask 1 and 6th out of 49 teams participating in subtask 2.

1 Introduction

The effect of Patronizing and Condescending Language (PCL) towards vulnerable communities in the media is not always conscious and the intention of the author is often to help the person or group they refer to (e.g. by raising awareness or funds or moving the audience to action). However, these superior attitudes and discourse of pity can routinize discrimination and make it less visible. While there has been substantial work on modeling language that purposefully undermines others, the modeling of PCL is still an emergent area of study in NLP since PCL is the speaker's unconscious superior speaking attitude, the special word that causes PCL is subtle compared to the keywords in other natural language processing problems.

The authors decided to evaluate the questions separately. In Semeval-2022 task 4: Patronizing and Condescending Language Detection(Pérez-Almendros et al., 2022), the purpose of subtask 1 is to identify whether a sentence is PCL. In contrast, the goal of subtask 2 is to indicate the presence of PCL at the text span level, which detects the exact categories in the seven categories of PCL.In subtask 1, the method of using data set retraining to make the pre-trained language model learn the specific distribution of the data set, adding keywords to the input, and integrating five RoBERTa-based models, subtask 2 is to select k from 7 For classification tasks, task-specific loss calculation methods are designed. These methods will be explained in detail in the following sections.

2 Background

Research on PCL has been in various fields such as language studies (Margić, 2017), sociolinguistics (Giles et al., 1993), politics (Huckin, 2002) or medicine (Komrad, 1983). In recent years, natural language processing systems for recognizing PCL languages have also begun to emerge, for example, (Wang and Potts, 2019) introduced the task of modeling humility in direct communication from an NLP perspective, and developed a dataset of annotated social media messages. In the same year, (Sap et al., 2019) discuss the social and power implications behind the use of certain languages, an important concept in the imbalanced power relations that often arise in condescending treatment. But there has not been a standard in terms of accuracy and definition of PCL. Therefore, this article will first explain the definition of PCL and define some categories of the linguistic techniques used to express PCL.

2.1 What is PCL

Somebody is patronizing or condescending when their language denotes a superior attitude towards

Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022), pages 438 - 444

others, talks down to them, or describes them or their situation in a charitable way, raising a feeling of pity and compassion. For example,*People across Australia ordered pizzas to be delivered on Saturday night, with the ample leftovers donated to local homeless shelters.* is a sentence that contains PCL for the sentence conveys a superior attitude towards the homeless.

Patronizing and Condescending Language (PCL) is often involuntary and unconscious, and the authors using such language are usually trying to help communities in need by e.g., raising awareness, moving the audience to action, or standing for the rights of the under-represented. On the other hand, due to its subtlety, subjectivity, and the (generally) good intentions behind its use, the audience is often unaware of this diminishing treatment. But PCL can potentially be very harmful, as it feeds stereotypes, routinizes discrimination, and drives to greater exclusion.

PCL detection is difficult both for humans and NLP systems, due to its subtle nature, its subjectivity, and the fair amount of world knowledge and commonsense reasoning required to understand this kind of language. With this task, we expect to push the boundaries of this new challenge in the NLP community.

2.2 Categories of PCL

Our PCL taxonomy has been defined based on previous works on PCL. We consider the following categories:

Unbalanced power relations The author distances themselves from the community or the situation they are talking about and expresses the will, capacity or responsibility to help those in need. It is also present when the author entitles themselves to give something positive to others in a more vulnerable situation, especially when what the author concedes is a right which they do not have any authority to decide to give.

Shallow solution A simple and superficial charitable action by the privileged community is presented either as life-saving/life-changing for the unprivileged one or as a solution for a deep-rooted problem.

Presupposition When the author assumes a situation as certain without having all the information or generalizes their or somebody else's experience as a categorical truth without presenting a valid, trustworthy source for it (e.g. a research work or survey). The use of stereotypes or clichés is also considered to be an example of presupposition.

Authority voice When the author stands themselves as a spokesperson of the group, or ex-plains or advises the members of a community about the community itself or a specific situation they are living.

Metaphor They can conceal PCL, as they cast an idea in another light, making a comparison between unrelated concepts, often with the objective of depicting a certain situation in a softer way. For the annotation of this dataset, euphemisms are considered as an example of metaphors.

Compassion The author presents the vulnerable individual or community as needy, raising a feeling of pity and compassion from the audience towards them. It is commonly characterized by the use of flowery wording that does not provide information, but the author enjoys the detailed and poetic description of the vulnerability.

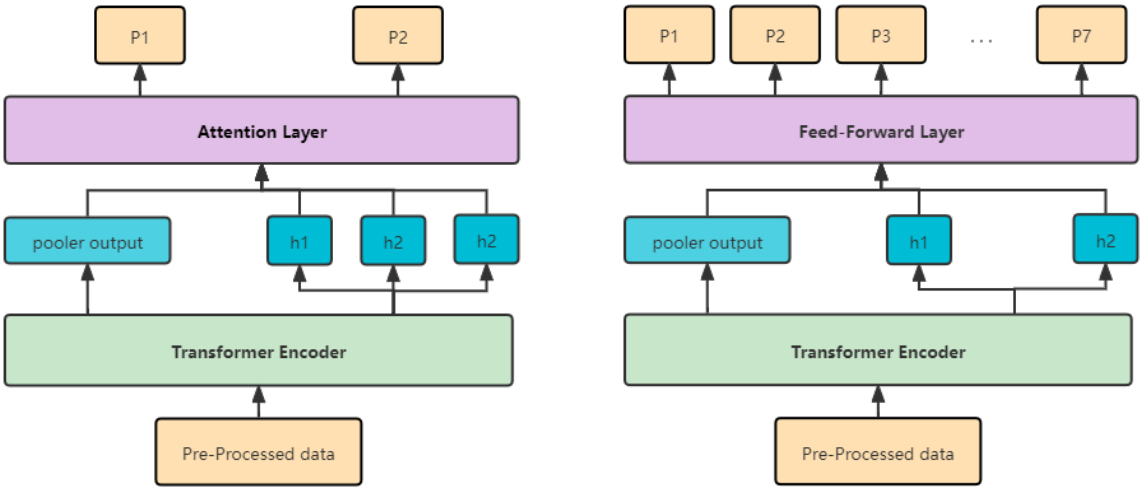
The poorer, the merrier The text is focused on the community, especially on how the vulnerability makes them better (e.g. stronger, happier, or more resilient) or how they share a positive attribute just for being part of a vulnerable community. People living in vulnerable situations have values to admire and learn from. The message expresses the idea of vulnerability as something beautiful o or poetic. We can think of the typical example of 'poor people are happier because they don't have material goods.

3 System description

In subtask 1 and subtask 2, we ensemble several models to obtain the results, which are all in RoBERTa-Based architecture (Liu et al., 2019). RoBERTa learns an inner representation of the English language that can be used to extract features useful for downstream tasks. In subtask 1, we pretrain the model on task-specific data. In subtask 2, we utilize multi-label categorical cross-entropy loss to improve performance.

3.1 Data pre-processing

Data for both subtask 1 and subtask 2 contain important information such as the keyword of the sentence and country code. In subtask 1, We truncate the original text centered on the keyword and



(a) RoBERTa-Based architecture of subtask 1.

(b) RoBERTa-Based architecture of subtask 2.

Figure 1: RoBERTa-Based architecture. In our system, the transformer encoder is RoBERTa, the input is pre-processed data mentioned above, pooler output denotes the last layer hidden-state of the first token of the sequence (classification token), hN means hidden states of the keyword extracted during pre-processing, which are the output of the Nth layer from the bottom of the encoder. Attention Layer uses the attention mechanism and calculates attention scores of inputs as weights. FeedForward Layer consists of two linear layers and performs the nonlinear transformation. PN denotes the probability that the Nth label belongs to the sentence.

extract the keyword and its position in the sentence. Also, the article location of the sentence, the keyword of the sentence and the country of the sentence are added to the input as additional features to make the model learn more useful information. Noting that the given country names are in abbreviated form, we restore them to their full form. With this approach, the input formats are shown in Tabel 1. What's more, considering the label imbalance problem of subtask 1, we find the sentences containing PCL from the data in subtask 2 and merge them to form several new sentences as data augmentation.

In subtask 2, We collect different labels of the same sentence to form a single piece of data and use the same way as subtask 1 to pre-process the data. Finally, we lowercase the pre-processed text of both subtasks before they are tokenized.

3.2 Task-Adaptive pretraining

It is proved that Task-Adaptive pretraining can help improve the performance of downstream tasks (Gururangan et al., 2020). In order to make our model better learn the distribution of the data for this task, we pretrain RoBERTa-large model on unlabeled data from subtask 1 and subtask 2. For the same consideration, we process the pretraining data in the way mentioned in Section 3.1.

We apply masked language modeling to pretrain RoBERTa model and use dynamic masking according to the RoBERTa paper. Compared with the original model, the model pretrained in this way can improve the performance to a greater extent.

3.3 RoBERTa-Based architecture

We tried different pretrained models on two subtasks. In our experiments, models initialized with RoBERTa outperform other models. So we choose RoBERTa and pretrain it on task-specific data as our basic model.

Model of subtask 1 In subtask 1, our system uses ensembles of 5 models based on pretrained RoBERTa-Based architecture. As shown in Figure 1(a), the RoBERTa-Based architecture consists of two components: Transformer Encoder, Attention Layer.

First, we pre-process the data to carry more information and tokenize the input into a form accepted by the model. The transformer encoder then is used to extract context representation of the whole sentence. During pretraining, transformer-based language models always use inputs with special tokens(such as [CLS]), so we take out the

extra information	Original text	Pre-processed text
par_id: 1964 keyword: refugee country: my	hospitals fill as rohingya refugees shiver through winter.	from 1964, keyword: refugee, country: Malaysia, hospitals fill as rohingya refugees shiver through winter.
par_id: 4136 keyword: homeless country: za	durban 's homeless communities reconciliation lunch.	from 4136, keyword: homeless, country: South Africa, durban 's homeless communities reconciliation lunch.

Table 1: Examples of Pre-processed text, where "extra information" means additional information in the training data, "Original text" means the original sentence to be judged as PCL or not, "Pre-processed text" means the sentence after pre-processing.

last layer hidden-state of the first token of the sequence(named pooler output), which is the representation of "[CLS]", to obtain a vector representation of the whole sentence. Also, we extract the hidden-state of the keyword of the last three layers, as it is proved that high-level network of transformer encoder learns rich semantic information features(Jawahar et al., 2019).

After we get the pooler output and hidden states of the keyword of each sentence, the two representations are concatenated and fed into an attention layer. We utilize the self-attention mechanism to calculate attention scores as weights in order to make the model attend to essential information. Finally, perform a linear transformation to get reduced representations. The whole process for the model to get the classification results is as follows:

$$Attn(e) = Softmax(A(eW_1+b_1)W_2+b_2) \quad (1)$$

$$Out = (Attn(e) \cdot e)W_3 + b_3 \quad (2)$$

Where e denotes the concatenation of pooler output and the last three hidden states of the keyword of each sentence. A is the Gaussian Error Linear Unit (GELU) (Hendrycks and Gimpel, 2016) activation function, Out denotes the probabilities of sentence-level labels.

Model of subtask 2 In subtask 2, our system uses ensembles of 2 models based on pretrained RoBERTa-Based architecture. As shown in Figure 1(b), the RoBERTa-Based architecture consists of two components: Transformer Encoder, Feed-Forward Layer.

The pre-processed data is obtained by the same method as subtask 1. We also take out the pooler output of the encoder, Furthermore, through experiments, we find that the last two hidden layer outputs of the keyword in each sentence are more effective for this subtask. Then the two representations are concatenated and fed into the Feed-Forward Layer, which is a combination of multiple linear and nonlinear transformations. Finally, we get the probability of each PCL category implied in the sentence.

According to a previous work (Sun et al., 2020), we use the loss function called multi-label categorical cross-entropy. Considering that the task is a multi-label classification problem, a common implementation is to use sigmoid activation, and then turn it into n binary classification problems, using the sum of the cross-entropy of the binary classification as the loss. Supposing k target categories are selected from n candidate categories, when n $\gg$ k, this approach will face a serious class imbalance problem. Therefore, we try to extend softmax and cross-entropy to multi-label classification, which expects each target class score is not less than the score for each non-target class. Instead of turning multi-label classification into multiple binary classification problems, it becomes a pairwise comparison of target class scores and non-target class scores to avoid class imbalance phenomenon. In the implementation, the weight of each label is automatically balanced with the good properties of log-sum-exp. The calculation process of the loss is as follows:

$$log(1 + \sum_{i \in \Omega neg} e^{s_i}) + log(1 + \sum_{i \in \Omega pos} e^{-s_j}) \quad (3)$$

Where Ωneg is the set of negative labels and Ωpos is the set of positive labels. s_N is the score of the Nth label in the corresponding set. In our experiments, we find that using this loss function can help improve the model performance.

task		Train	Valid	Total
subtask 1	PCL	7581	794	2094
	not PCL	1895	199	8375
subtask 2	Unb. power rel.	574	142	716
	Shallow solution	160	36	196
	Presupposition	162	62	224
	Authority voice	192	38	230
	Metaphor	145	52	197
	Compassion	363	106	469
	The p., the mer.	29	11	40

Table 2: Statistics of the dataset

4 Experiment

4.1 Dataset

We trained our models on SemEval-2022 Task 4 training data which is an annotated dataset with Patronizing and Condescending Language(PCL) towards vulnerable communities(Pérez-Almendros et al., 2020). The organizers not only annotated all text spans as containing PCL, but also provided PCL category labels, including a total of seven more fine-grained level categories. At last, there are 10469 marked data in total. The organizers of the competition have divided the data into training set and validation set using the split ratio 8:2. And each text contains an average of 232 tokens. Subtask 1 is a binary classification task, so the labels are just PCL and not PCL, but most of them contain PCL accounts for the majority resulting in imbalance between classes. Subtask 2 is a multi-label binary classification task that aims to predict which PCL categories these texts belong to. And The proportion of each category is more balanced. The statistics of these datasets are given in Table 2.

4.2 Metric

For Subtask 1, it is a binary classification task that will be evaluated using F1 value of the positive class. Subtask 2 is a multi-label classification task, which is evaluated by macro-F1. The calculation formula is as follows. The experimental results are all obtained by averaging three runs with different random initialization.

$$F1 = \frac{2 \times P \times R}{P + R} \tag{4}$$

$$Macro - F1 = \frac{1}{n} \sum_{s=1}^{n} F_z \tag{5}$$

4.3 Experiment Settings

After many experiments, the results show that the effect of using RoBERTa-large is the best and most stable. So at last, all models used in the end are all based on the RoBERTa-large. At the same time, the maximum length of the text is 512. We use Adam as the optimizer with a learning rate of 2e-5. We also use gradual warmup(Goyal et al., 2017) and cosine annealing schedule for learning rate. The coefficient of L2-regularization is 1e-5 and batch size is 32. All the experiments are done on 2 NVIDIA 3090 GPUs, and Limited by the size of GPU memory, we used gradient accumulation.

4.4 Results

For Subtask 1, we designed four models in total: (1)**RoBERTa-ft**: simply fine-tune RoBERTa-large model;(2) **RoBERTa-cls3**: extract the first hidden vector of the last three layers of RoBERTa model and cat them, then pass through a self-attention layer. At last, the hidden vector obtained by multiplying the softmax weight is classified through the linear layer to get predictions;(3) **RoBERTa-cls4**: similar with model1, The only difference is that this model extracts the first hidden vector of the last four layers. (4) **RoBERTa-key**: this model will take out the hidden vector corresponding to the keyword and splice it with the pooler-out vector, then pass through the linear layer to get predictions. For Subtask 2, we build two models at last, including: (1) **RoBERTa-ff**: cat the hidden vectors of the last two layers, and then spliced with pooler-out to pass through a feedforward layer. (2)**RoBERTa-att**: cat the hidden vectors of the last two layers, and then spliced with pooler-out to pass through an attention layer.

Table 3 shows the best F1 values of each model on the official Subtask 1 validation set. And Table 4 shows the F1 values of the above two models on each category and their average values in Subtask 2. For both Subtask 1 and Subtask 2, we set the maximum number of epochs to 10 and open early stop.

We can see from the data in the table that all of the considered methods clearly outperform the baseline. For Subtask 1, the RoBERTa-key achieves the best performance. We also try some other methods, such as extracting the last four hidden vectors of the model, calculating the average value or the maximum value, but their effect is not as good as the above methods. We think that

Method	acc	F1
baseline	-	0.5211
RoBERTa-ft	0.9254	0.6385
RoBERTa-cls3	0.9288	0.6410
RoBERTa-cls4	0.9303	0.6439
RoBERTa-key	0.9298	**0.6475**

Table 3: Results of detecting PCL, viewed as a binary classification problem (Subtask 1).

	Unb.	Auth.	Sha.	Pre.	Com.	Meta.	The p.	average
method	F1	F1	F1	F1	F1	F1	F1	F1
Baseline	0.3844	0.3614	0.3212	0.3745	0.3187	0.376	0.1045	0.3201
Robeta-att	0.5876	0.5423	0.4224	0.4341	0.4359	0.5026	0.1635	0.4412
RoBERTa-ff	0.5958	0.4942	0.3942	0.4492	0.3971	0.4874	0.2887	**0.4438**

Table 4: Results for the problem of categorizing PCL, viewed as a multi-label classification problem (Subtask 2).

Model	F1
RoBERTa	0.5921
+Prefix template	0.6045
+key-hidden	0.6127
+pre-train	0.6386
Last	**0.6475**

Table 5: Ablation results of our model

the keyword can be regarded as an object. For example, if the keyword is poor, the passage is to judge whether the author has an arrogant attitude towards the poor. Therefore, the hidden vector corresponding to the keyword contains more feature information and is very helpful for our judgment. At the same time, extracting the last few hidden vectors contains more information with different granularity. For Subtask 2, We also tried many other different structures, the RoBERTa-ff achieves the best performance we find after a lot of experiments.

4.5 Ablation

We used some stricks and methods in the competition, and we show the improvement effect of each method through the ablation experimental results on Subtask 1 in Table 5. It can be seen that the improvement brought by pre-training is the most significant which improves by more than two points. The second is to add a prefix template. In addition to these methods in the table, there are also slight improvements by some stricks such as resampling.

5 Conclusion

The paper describes our system at SemEval-2022 Task 4, which uses several different models based on RoBERTa. We used a series of methods such as pre-training, constructing prefix templates, and model fusion to achieve relatively good results. As we can see, using RoBERTa as the network backbone achieves better performance in this task. Also post-training and using the hidden vectors of the last few layers of RoBERTa can improve the effect. At the same time, we also tried FGM, focal loss and other methods, while none of them seemed to be beneficial for our task. Still, the F1 value is less than satisfactory, so we can see that identifying and categorizing Patronizing and Condescending Language are difficult challenges. In the future, we will consider using some external knowledge to help the judgment of the model.

References

Howard Giles, Susan Fox, and Elisa Smith. 1993. Patronizing the elderly: Intergenerational evaluations. *Research on Language and Social Interaction*, 26(2):129–149.

Priya Goyal, Piotr Dollár, Ross Girshick, Pieter Noordhuis, Lukasz Wesolowski, Aapo Kyrola, Andrew Tulloch, Yangqing Jia, and Kaiming He. 2017. Accurate, large minibatch sgd: Training imagenet in 1 hour. *arXiv preprint arXiv:1706.02677*.

Suchin Gururangan, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A Smith. 2020. Don't stop pretraining: adapt language models to domains and tasks. *arXiv preprint arXiv:2004.10964*.

Dan Hendrycks and Kevin Gimpel. 2016. Gaussian error linear units (gelus). *arXiv preprint arXiv:1606.08415*.

Thomas Huckin. 2002. Critical discourse analysis and the discourse of condescension. *Discourse studies in composition*, 155:176.

Ganesh Jawahar, Benoît Sagot, and Djamé Seddah. 2019. What does bert learn about the structure of language? In *ACL 2019-57th Annual Meeting of the Association for Computational Linguistics*.

Mark S Komrad. 1983. A defence of medical paternalism: maximising patients' autonomy. *Journal of medical ethics*, 9(1):38–44.

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. *arXiv preprint arXiv:1907.11692*.

Branka Drljača Margić. 2017. Communication courtesy or condescension? linguistic accommodation of native to non-native speakers of english. *Journal of English as a lingua franca*, 6(1):29–55.

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.

Maarten Sap, Saadia Gabriel, Lianhui Qin, Dan Jurafsky, Noah A Smith, and Yejin Choi. 2019. Social bias frames: Reasoning about social and power implications of language. *arXiv preprint arXiv:1911.03891*.

Yifan Sun, Changmao Cheng, Yuhan Zhang, Chi Zhang, Liang Zheng, Zhongdao Wang, and Yichen Wei. 2020. Circle loss: A unified perspective of pair similarity optimization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6398–6407.

Zijian Wang and Christopher Potts. 2019. Talkdown: A corpus for condescension detection in context. *arXiv preprint arXiv:1909.11272*.