shefnlp at SemEval-2023 Task 10: Compute-Efficient Category Adapters

Thomas Pickard and **Tyler Loakman** and **Mugdha Pandya**
The University of Sheffield
Sheffield, UK
{tmrpickard1, tcloakman1, mugdha.pandya}@sheffield.ac.uk

Abstract

As social media platforms grow, so too does the volume of hate speech and negative sentiment expressed towards particular social groups. In this paper, we describe our approach to SemEval-2023 Task 10, involving the detection and classification of online sexism (abuse directed towards women), with fine-grained categorisations intended to facilitate the development of a more nuanced understanding of the ideologies and processes through which online sexism is expressed. We experiment with several approaches involving language model finetuning, class-specific adapters, and pseudo-labelling. Our best-performing models involve the training of adapters specific to each subtask category (combined via fusion layers) using a weighted loss function, in addition to performing naive pseudo-labelling on a large quantity of unlabelled data. We successfully outperform the baseline models on all 3 subtasks, placing 56th (of 84) on Task A, 43rd (of 69) on Task B, and 37th (of 63) on Task C.

1 Introduction

Sexism, as defined within this task (Kirk et al., 2023), refers to any abuse or negative sentiment that is directed towards women based on their gender, or based on their gender combined with one or more other identity attributes. Whilst sexism may, in the general case, be gender agnostic (i.e. also applies to discrimination towards men or any other identified gender), it remains true that this form of discrimination disproportionately targets women. As the userbases of social media platforms grow, and disputes regarding the censorship and moderation of particular opinions and ideologies increase, extreme negative opinions towards particular groups are allowed to proliferate. Consequently, much recent research in NLP has been on the detection of various forms of online hate, including racism, homophobia, and sexism, to combat its growing prevalence (Alkomah and Ma, 2022). Within SemEval2023 Task 10 (Kirk et al., 2023), the 3 component subtasks relate to increasing levels of granularity in the classification of sexism in short English texts sourced from Reddit and Gab. Task A presents a binary detection task to determine whether social media posts express sexism, whilst Task B splits this further into a 4-way multi-class classification task regarding broad categories of sexist content (threats, derogation, animosity, and prejudiced discussions). Finally, Task C presents the most fine-grained level of classification, presenting an 11-class taxonomy (described as "vectors of sexism") which further splits the 4 classes from Task B into specific constituents.

2 Background

Recent years have witnessed many works in the area of hate speech and offensive language detection. As the field develops, research is beginning to diverge from simple offence detection into more fine-grained classification into areas such as racism, homophobia, and sexism, thereby getting closer to identifying not only whether language is hateful, but also why (Zia et al., 2021). In pursuit of this goal, there have been a range of previous shared tasks focusing on such detection, including most recently TRAC-2 (Kumar et al., 2020) and MAMI (Fersini et al., 2022). In becoming more precise in identifying types of hate speech, problems have been shown in regard to annotator agreement concerning offensive language, with the addition of more categories consequently increasing uncertainty and levels of disagreement across human raters, demonstrating the increasing difficulty of the 3 subtasks in this current SemEval 2023 challenge (Sap et al., 2022; Larimore et al., 2021). The winning systems from such tasks frequently employ ensemble learning involving the training of multiple separate models (Zhang and Wang, 2022). In contrast to this, we focus on approaches that are more compute-efficient, primarily via the use

Proceedings of the The 17th International Workshop on Semantic Evaluation (SemEval-2023), pages 1069–1075

of adapters (Houlsby et al., 2019; He et al., 2021; Pfeiffer et al., 2021) and weighted loss to handle class imbalances.

3 System Overview

As described in Kirk et al. (2023), the task data for each task were divided into training, development and test sets in a 70:10:20 ratio. We further subdivided the training data, reserving 10% for model validation; within each experiment, model checkpoints were saved following each training epoch and their performance on this validation set calculated. The best-performing checkpoint (highest macro-F1) was then restored. A further 10% of the training data was reserved for comparison between models trained under different experimental conditions. For clarity, performance results throughout this paper are reported for this selection set (**Sel**, 7% of the original task data), the organisers' development set (**Dev**, 10%) and, for our final submitted systems, the task test set (**Final**, 20% of the original task data).

Most of our experiments (and all of our submitted systems) made use of adapters, which are lightweight, modular extensions to pre-trained transformer language models introduced by Houlsby et al. (2019). Adapters consist of small modules inserted between the layers of a transformer model. The pre-trained network weights are frozen and the adapter weights initialised close to unity before they are trained, effectively enabling tuning of the information flow to suit a particular task. This configuration reduces the number of parameters which must be trained when compared with full model fine-tuning, and can be useful in a multi-task learning set-up, with distinct adapters suited to each task sharing identical pre-trained model weights. Adapters may be combined in a number of structures, including parallel and sequential configurations (Pfeiffer et al., 2021). Our experiments made use of the AdapterHub framework (Pfeiffer et al., 2020) to configure and train adapters. AdapterHub is a fork of the Huggingface Transformers library (Wolf et al., 2020) with adapter-specific functionality, and also offers an online platform for sharing of trained adapter modules.

3.1 Task A

For Task A, our approach was based on the `RoBERTa-base` architecture (Liu et al., 2019) as restored from a publicly available checkpoint from HuggingFace.[1] This model was chosen firstly because it provided a performance improvement over the task baseline models, and secondly in order to allow us to experiment using the Emotion Adapters of Poth et al. (2021), who employed the RoBERTa architecture to train adapters for emotion classification of Twitter data. Our `RoBERTa-base` baseline model achieved a macro average F1 score of 0.829 on our selection set and 0.832 on the held-out task dev set.[2]

Our final submitted system froze all weights in the `RoBERTa-base` model, and inserted two sets of adapters (combined via adapter fusion) and a classification head.[3] The parameters for these were trained using a weighted loss function (see §4.2). The system achieved macro average F1 of 0.829 on the selection set and 0.818 on the task dev set. This is comparable with our baseline model results, while requiring training of significantly fewer parameters and increasing system portability.

3.2 Task B

For Task B, our system again employed a `RoBERTa-base` model with frozen parameter weights, to which we added a set of adapters and a classification head. This configuration achieved similar performance on our selection set to our fine-tuned `RoBERTa-base` baseline, and outperformed it on the competition dev set, while involving computation of far fewer model weights.

Given the hierarchical nature of the class labelling for the task, it made intuitive sense that information learned by the model when classifying items as sexist or not in a binary setup (Task A) might be beneficial in predicting the sub-labels. To explore this approach, we took class-specific adapters trained for Task A and fed their outputs into further adapters which were then trained for Task B. This 'stacked' architecture proved effective on our selection set but performed less well on the competition dev data than our final submitted system (see Table 2). It could be valuable to explore this idea further, especially in combination with pseudo-labelling or other techniques to increase the quantity of available training data.

[1] https://huggingface.co/roberta-base

[2] Baseline models were trained as in the task description: the prediction head is trained and other parameters are fine-tuned on 80% of the labelled training data for up to 30 epochs.

[3] Other than variation in initialisation and shuffling of training data, the two adapters were trained in the same fashion.

3.3 Task C

Our Task C system combined a number of refinements, intended to reduce the impact of the class imbalance and limited labelled training data:

- **Class Adapters**: We trained adapters for each of the 11 class labels (using binarised labels, e.g. to discriminate between '2.2 aggressive and emotive attacks' and all other Task C labels). These were fused together and a combined classification head was trained for the multi-label classification task.
- **Weighted Loss**: We weight the per-class loss in inverse proportion to the frequency of that class in the training data.
- **Pseudo-labelling**: In order to increase the volume of training data available, we implemented pseudo-labelling of the additional unlabelled Gab and Reddit data provided by the competition organisers. A `DistilBERT` (Sanh et al., 2019) model was restored from a public checkpoint[4] and further pre-trained (with a masked language modelling objective) for 30 epochs on the unlabelled data, with a view to adapting the model to better suit the domain. This model was then fine-tuned on the labelled data to predict the class labels and then used to predict labels for the unlabelled datasets.[5] 3% of these pseudo-labelled items (randomly sampled) were used to augment the human-labelled data for training our system. This sample size meant that the volume of human- and pseudo-labelled training examples were approximately equal.

On our model selection dataset, this system performed better than any of the other configurations we analysed, with a macro-average F1 score of 0.41. While its performance on the competition dev dataset was similar (F1 0.42), it is worth noting that a less involved setup (adapters trained solely on the labelled data with an increased learning rate, equivalent to our submission system for Task B) obtained a significantly higher score (F1 0.46) on the dev data. These observations suggest that the details of the test data used may have a noticeable impact on model performance metrics, particularly for Task C. This is perhaps unsurprising given the relatively small volume of labelled data and large number of highly-imbalanced class labels.

[4] https://huggingface.co/distilbert-base-uncased

[5] The classification performance of this system is presented in Table 3 as 'Baseline 4b'.

4 Experimental Setup

4.1 Data Preparation

The data provided for this task by the organisers (Kirk et al., 2023) on which we trained our models consisted of 14,000 isolated posts from Reddit and Gab (a far-right social media platform) that were labelled for the 3 subtasks by trained human annotators. In total, the following entries are available for each post:

- **rewire_id**: A unique identifier for each dataset entry.
- **text**: the pre-processed text of each post.
- **label_sexist**: The binary classification of each post into *sexist* or *not sexist*.
- **label_category**: The further breakdown into 4 categories of sexism – "1. threats, plans to harm and incitement", "2. derogation", "3. animosity", and "4. prejudiced discussions". Additionally, a *none* label is used for those entries which were deemed to not exhibit sexism in the binary case.
- **label_vector**: A further breakdown of the 4 *label_category* distinctions into 2-4 subcategories (e.g. "3.1 Casual use of gendered slurs, profanities, & insults" and "3.2 Immutable gender differences and gender stereotypes" as two sub-categories for "3. Animosity").

A development (dev) set of a further 2,000 items was also supplied, without labels. Performance metrics on this set supplied by the competition organisers were used in our model selection.

Additionally, 1 million unlabelled entries were provided for both Reddit and Gab sources (2 million total), with text pre-processing applied in the same manner as the labelled data. We used this data for experiments involving continued model pre-training, and a subset of it for pseudo-labelling.

As described above (§3), 10% of the training dataset was reserved for evaluation of model performance at the end of each experimental training epoch, and a further 10% for model selection by comparison between different experimental configurations.

Pretrained Model	Configuration	Training	Params	F1 (Macro Average)		
				Sel	Dev	Final
DistilBERT	Baseline	Fine-tune on labelled data	α	0.8129	0.7898	
RoBERTa	Baseline	Fine-tune on labelled data	α	0.8294	**0.8325**	
RoBERTa	Emotion adapters	Retrain adapters	α	0.8214	0.8026	
RoBERTa	Custom adapters		α	0.8357	0.8018	
RoBERTa	Per-class adapters + fusion	Weighted loss	α	0.8286	0.8182	0.8181
RoBERTa	Per-class adapters + fusion	Oversampled minority classes	α	0.8220	0.8230	
RoBERTa	Custom adapters	Hyperparameter tuned	β	0.8336	**0.8303**	
DistilBERT	Baseline 4b	Further pretraining	α	0.8298	0.8076	
DistilBERT	Per-class adapters + fusion	Weighted loss, pseudo-labelling	β	0.8363	0.8122	

Table 1: Experimental configurations and results, Task A. See §5 for details.

Pretrained Model	Configuration	Training	Params	F1 (Macro Average)		
				Sel	Dev	Final
DistilBERT	Baseline	Fine-tune on labelled data	α	0.5366	0.5792	
RoBERTa	Baseline	Fine-tune on labelled data	α	0.6009	**0.6153**	
RoBERTa	Emotion adapters	Retrain adapters	α	0.5369	0.5721	
RoBERTa	Custom adapters		α	0.5059	0.5837	
RoBERTa	Per-class adapters + fusion	Weighted loss	α	0.5812	0.6155	
RoBERTa	Per-class adapters + fusion	Oversampled minority classes	α	0.5453	0.5708	
RoBERTa	Adapter stack	Weighted loss	α	0.6121	0.6035	
RoBERTa	Custom adapters	Hyperparameter tuned	β	0.5965	**0.6201**	0.5890
DistilBERT	Baseline 4b	Further pretraining	α	0.5556	0.5923	
DistilBERT	Per-class adapters + fusion	Weighted loss, pseudo-labelling	β	0.5404	0.5875	
DistilBERT	Adapter stack	Weighted loss, pseudo-labelling	β	0.5525	0.5885	

Table 2: Experimental configurations and results, Task B. See §5 for details.

4.2 Model Training

Each of our final models was trained using a weighted loss function, in which the per-class loss was weighted in inverse proportion to the frequency of that class in the training data. This was done in order to reduce the impact of class imbalance on the model. Our adapter configuration and training was performed using AdapterHub (Pfeiffer et al., 2020).

4.2.1 Task A

Following the Task A submission deadline, we performed a hyperparameter tuning exercise on our submitted system using Optuna (Akiba et al., 2019), and were able to increase the F1 scores to 0.834 / 0.830; this version of the system is closely comparable in terms of performance with a fully-fine-tuned baseline model. Most notably, the optimal learning rate was 2.6e^{-4}, more than a factor of 10 greater than the rate used in the baseline system. This aligns with previous findings on transformer adapters which have found that they benefit from higher learning rates (Pfeiffer et al., 2021).

4.2.2 Tasks B and C

The hyperparameters used to train the model were found through optimisation for the Task A configuration – due to time constraints, we were unable to separately optimise training parameters for Tasks B and C. As noted above, the adapter training benefited from the use of a relatively high LR.

5 Results

The results of all experiments we conducted are shown in tables 1, 2 and 3.

Two distinct hyperparameter configurations were used, the latter having been obtained from a hyperparameter search for Task A:

- α : Batch size 32, learning rate 2e^{-5}, Adam epsilon 1e^{-8}.
- β : Batch size 64, learning rate 2.6528e^{-4}, Adam epsilon 1.3776e^{-8}.

All training used the Adam optimizer (Kingma and Ba, 2014) and each model was trained for at least 20 epochs on a single GPU, with checkpoints after each epoch. The checkpoint which delivered

Pretrained Model	Configuration	Training	Params	F1 (Macro Average)		
				Sel	Dev	Final
DistilBERT	Baseline	Fine-tune on labelled data	α	0.3638	0.3647	
RoBERTa	Baseline	Fine-tune on labelled data	α	0.4011	**0.4220**	
RoBERTa	Emotion adapters	Retrain adapters	α	0.2893	0.2982	
RoBERTa	Custom adapters		α	0.3106	0.2850	
RoBERTa	Per-class adapters + fusion	Weighted loss	α	0.3374	0.3346	
RoBERTa	Per-class adapters + fusion	Oversampled minority classes	α	0.3366	0.3316	
RoBERTa	Adapter stack	Weighted loss	α	0.3708	0.3702	
RoBERTa	Custom adapters	Hyperparameter tuned	β	0.3952	**0.4613**	
DistilBERT	Baseline 4b	Further pretraining	α	0.3697	0.4026	
DistilBERT	Per-class adapters + fusion	Weighted loss, pseudo-labelling	β	0.4104	0.4189	0.3811
DistilBERT	Adapter stack	Weighted loss, pseudo-labelling	β	0.3827	0.4176	

Table 3: Experimental configurations and results, Task C. See §5 for details.

the highest macro-F1 score on our validation set (10% of the training data) was then evaluated.

Macro average F1 scores are reported for our model selection set (10% of the training data), the competition development set and the final results reported by the organisers on the task test set (where applicable).

All models are `-base` configurations. Emotion adapters per Poth et al. (2021). Further pretraining, where listed, was carried out with masked language modelling objective on the unlabelled training data and followed by fine-tuning for classification on the labelled set. Adapter stack configurations use class adapters for the preceding subtask(s) stacked in serial with those trained for this subtask, with a final fusion and prediction layer.

Overall, in Task A our system achieved an F1 of 0.8181 (56$^{\text{th}}$ of 84 submissions), whilst for Task B we achieve an F1 of 0.5890 (43$^{\text{rd}}$ of 69 submissions), and for Task C we achieve an F1 of 0.3811 (37$^{\text{th}}$ of 63 submissions), according to the final ranking results provided by the organisers.

Our results on the competition development set suggest that these performance outcomes are comparable to fine-tuning a `RoBERTa-base` model on the labelled task data, but are achieved by training only the much more lightweight adapters; this approach proved competitive through less intensive training and yielding a more portable system.

6 Conclusion

In this paper, we demonstrate the methods used in our submission towards the 3 subtasks in the EDOS 2023 SemEval challenge. Specifically, we demonstrate the effectiveness of class-specific adapters and fusion layers for improving classification performance in setups where some data classes are imbalanced, in addition to the use of weighted loss for further handling these imbalances. We hope that the importance of detecting content such as sexism is appreciated by the wider community, and that further shared tasks are developed to allow more rapid progress to facilitate content moderation. We additionally hope that the approaches demonstrated here promote further use of more compute-efficient methods in shared task submissions.

While they did not yield the highest performance on the competition development set, our adapter stacking experiments showed promise, and we suspect that there may be value in further refining this approach for hierarchical classification problems in the future.

In recent years, some research has criticised the use of gold standard labels for subjective tasks like sexism detection, hate speech and offensive language detection, sarcasm detection, etc. Annotator disagreements in such tasks often occur due to people having differing, valid opinions. The removal of these variations in opinion to establish a single 'gold standard' label could introduce bias into these labelled datasets. In the future, we will explore the effect of inter-annotator disagreement on bias in sexism detection.

Acknowledgements

This work was partially supported by the Centre for Doctoral Training in Speech and Language Technologies (SLT) and their Applications funded by UK Research and Innovation [grant number EP/S023062/1].

This work was partially supported by the ESRC-funded project Responsible AI for Inclusive, Democratic Societies [grant number R/163157-11-1].

References

Takuya Akiba, Shotaro Sano, Toshihiko Yanase, Takeru Ohta, and Masanori Koyama. 2019. Optuna: A next-generation hyperparameter optimization framework. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*.

Fatimah Alkomah and Xiaogang Ma. 2022. A literature review of textual hate speech detection methods and datasets. *Information*, 13(6):273.

Elisabetta Fersini, Francesca Gasparini, Giulia Rizzi, Aurora Saibene, Berta Chulvi, Paolo Rosso, Alyssa Lees, and Jeffrey Sorensen. 2022. SemEval-2022 task 5: Multimedia automatic misogyny identification. In *Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022)*, pages 533–549, Seattle, United States. Association for Computational Linguistics.

Ruidan He, Linlin Liu, Hai Ye, Qingyu Tan, Bosheng Ding, Liying Cheng, Jiawei Low, Lidong Bing, and Luo Si. 2021. On the effectiveness of adapter-based tuning for pretrained language model adaptation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 2208–2222, Online. Association for Computational Linguistics.

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin de Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. 2019. Parameter-efficient transfer learning for nlp.

Diederik P. Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization.

Hannah Rose Kirk, Wenjie Yin, Bertie Vidgen, and Paul Röttger. 2023. SemEval-2023 Task 10: Explainable Detection of Online Sexism. In *Proceedings of the 17th International Workshop on Semantic Evaluation*, Toronto, Canada. Association for Computational Linguistics.

Ritesh Kumar, Atul Kr. Ojha, Shervin Malmasi, and Marcos Zampieri. 2020. Evaluating aggression identification in social media. In *Proceedings of the Second Workshop on Trolling, Aggression and Cyberbullying*, pages 1–5, Marseille, France. European Language Resources Association (ELRA).

Savannah Larimore, Ian Kennedy, Breon Haskett, and Alina Arseniev-Koehler. 2021. Reconsidering annotator disagreement about racist language: Noise or signal? In *Proceedings of the Ninth International Workshop on Natural Language Processing for Social Media*, pages 81–90, Online. Association for Computational Linguistics.

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.

Jonas Pfeiffer, Aishwarya Kamath, Andreas Rücklé, Kyunghyun Cho, and Iryna Gurevych. 2021. AdapterFusion: Non-destructive task composition for transfer learning. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 487–503, Online. Association for Computational Linguistics.

Jonas Pfeiffer, Andreas Rücklé, Clifton Poth, Aishwarya Kamath, Ivan Vulić, Sebastian Ruder, Kyunghyun Cho, and Iryna Gurevych. 2020. Adapterhub: A framework for adapting transformers. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP 2020): Systems Demonstrations*, pages 46–54, Online. Association for Computational Linguistics.

Clifton Poth, Jonas Pfeiffer, Andreas Rücklé, and Iryna Gurevych. 2021. What to pre-train on? Efficient intermediate task selection. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 10585–10605, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter.

Maarten Sap, Swabha Swayamdipta, Laura Vianna, Xuhui Zhou, Yejin Choi, and Noah A. Smith. 2022. Annotators with attitudes: How annotator beliefs and identities bias toxic language detection. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5884–5906, Seattle, United States. Association for Computational Linguistics.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. 2020. Huggingface's transformers: State-of-the-art natural language processing.

Jing Zhang and Yujin Wang. 2022. SRCB at SemEval-2022 task 5: Pretraining based image to text late sequential fusion system for multimodal misogynous meme identification. In *Proceedings of the*

16th International Workshop on Semantic Evaluation (SemEval-2022), pages 585–596, Seattle, United States. Association for Computational Linguistics.

Haris Bin Zia, Ignacio Castro, and Gareth Tyson. 2021. Racist or sexist meme? classifying memes beyond hateful. In *Proceedings of the 5th Workshop on Online Abuse and Harms (WOAH 2021)*, pages 215–219, Online. Association for Computational Linguistics.