QMUL-SDS at EXIST: Leveraging Pre-trained Semantics and Lexical Features for Multilingual Sexism Detection in Social Networks

Aiqi Jiang and Arkaitz Zubiaga

Queen Mary University of London, London, UK
{a.jiang,a.zubiaga}@qmul.ac.uk

Abstract. Online sexism is an increasing concern for those who experience gender-based abuse in social media platforms as it has affected the healthy development of the Internet with negative impacts in society. The EXIST shared task proposes the first task on sEXism Identification in Social neTworks (EXIST) at IberLEF 2021 [30]. It provides a benchmark sexism dataset with Twitter and Gab posts in both English and Spanish, along with a task articulated in two subtasks consisting in sexism detection at different levels of granularity: Subtask 1 Sexism Identification is a classical binary classification task to determine whether a given text is sexist or not, while Subtask 2 Sexism Categorisation is a finer-grained classification task focused on distinguishing different types of sexism. In this paper, we describe the participation of the QMUL-SDS team in EXIST. We propose an architecture made of the last 4 hidden states of XLM-RoBERTa and a TextCNN with 3 kernels. Our model also exploits lexical features relying on the use of new and existing lexicons of abusive words, with a special focus on sexist slurs and abusive words targeting women. Our team ranked 11^{th} in Subtask 1 and 4^{th} in Subtask 2 among all the teams on the leaderboard, clearly outperforming the baselines offered by EXIST.

Keywords: Sexism Identification · Hate Speech Detection · Abusive Language Detection · Multilingual Text Classification · Social Network.

1 Introduction

Along with an unprecedented ability for communication and information sharing, social media platforms provide an anonymous environment which allows users to take aggressive attitudes towards specific groups or individuals by posting abusive language. This leads to increased occurrences of incidents, hostile behaviours and remarks of harassment [32,10,11,4]. Abusive language is one of the most important conceptual categories in anti-oppression politics today [14,32].

IberLEF 2021, September 2021, Málaga, Spain.

Gender-based speech is a common type of abusive language online which disparages an individual or group on the basis of their gender, currently considered as a deteriorating factor in social networks [13].

In the recent years, due to the increasing amount of user-generated content and the diversity of user behaviour towards women in social media, manual inspection and moderation of gender-related contents becomes unmanageable. The academic community has seen a rapid increase in research tackling the automatic detection of hateful behaviour towards women in both monolingual and multilingual scenarios, spreading across various social media platforms (such as Facebook and Twitter) [22,37]. The first attempt is made by Hewitt et al. [20] who explores the manual classification of English misogynous tweets, and the first survey of automatic misogyny identification in social media is conducted by Anzovino et al. [1]. Chowdhury et al. [15] aggregate experiences of sexual abuse to facilitate a better understanding of social media construction and Nozza et al. [33] attempt to measure and mitigate unintended bias in machine learning models for misogyny detection. An extensive of misogyny detection is then conducted especially in multilingual and cross-domain scenarios [34]. Since 2018, from the perspective of machine learning and computational linguistics, many international evaluation campaigns have been organised to identify online cases of multilingual abusive language against women, such as AMI@Evalita 2018 in English and Italian [10], AMI@IberEval 2018 in English and Spanish [12], HatEval@SemEval 2019 in English and Spanish [2], AMI@Evalita 2020 in Italian [11] and ArMI@HASOC 2021 [31] in Arabic.

However, most previous studies have concentrated on detecting misogynous behaviour online [41,1,13,34], while misogynous behaviour is not always equivalent to sexism. Misogyny frequently implies a hostile attitude with obvious hatred against women [34]. As for sexism, Glick and Fiske [16] define two forms of sexism: hostile sexism and benevolent sexism. Hostile sexism is characterised by an explicitly negative attitude towards women, while benevolent sexism is more subtle with seemingly positive characteristics. Sexism includes a wide range of behaviours (such as stereotyping, ideological issues, sexual violence, etc.) [29,1], and may be expressed in different ways: direct, indirect, descriptive or reported [19,5]. Thus, misogyny is only one case of sexism [29]. Hence, given subtle or implicit expressions of sexism, dealing with the detection of sexism in a wide spectrum of sexist attitudes and behaviours is necessary as these are, in fact, the most frequent and dangerous for society [36]. The purpose of the EXIST@IberLEF 2021 shared task [38,30] is to consider sexist behaviour in a broad sense, from explicit misogyny to other subtle behaviours involving implicit sexism. The EXIST dataset contains various types of sexist expressions and related phenomena, including descriptive or reported assertions, where a sexist post is a report or description of sexist behaviour.

More recently, general pre-trained language models (PLM) have shown their capacity to improve the performance of NLP systems for most tasks on canonical data. Among the recent work for multilingual PLMs, multilingual BERT (BERT) [9] and cross-lingual language model (XLM) [8] have stood out, thanks

to the effectiveness of pre-training large transformers on multiple languages at once in the field of cross-lingual understanding [39]. However, due to the limited availability of training corpora, XLM-RoBERTa model (XLM-R) [7] has become the new state-of-the-art (SOTA) multilingual PLM by extending the amount of training data and enlarging the length of sentences. These SOTA PLMs are usually fine-tuned to some downstream classification tasks, such as multilingual sexism detection [34], whereas few of them consider to induct external knowledge in a multilingual scenario into the model, such as linguistic information from a domain-specific lexicon.

Inspired by the work in [26], in this paper we propose a novel approach (XRCNN-Ex) by combining XLM-R [7] with a TextCNN [24] and infusing External lexical knowledge from HurtLex [3] to handle two subtasks of EXIST. Given the scarcity of semantic information in the commonly-used pooler output of XLM-R, XRCNN-Ex aggregates the last 4 hidden states of XLM-R to obtain the representations with ampler semantic features. Then we construct a TextCNN with 3 different kernels to capture various local features from XLM-R, which decreases the memory cost with a smaller number of parameters and proceeds a faster training speed with lower computation compared to those RNN-based models. Additionally, external knowledge from the domain-specific lexicon HurtLex is fed into the structure of XRCNN in order to investigate the effectiveness of lexical information on the performance. In our experimental and official results, the basic architecture XRCNN in our proposed model presents a notable achievement, while the performance of XRCNN-Ex is comparatively unstable and inferior in the final submission. We discuss this case in Section 5. When it comes to the team ranking, we ranked 11^{th} in subtask 1 sexism identification and 4^{th} in subtask 2 sexism categorisation. In submission ranking, we ranked 14^{th} (accuracy score of 0.761) and 5^{th} (macro f1 score of 0.559) respectively.

2 EXIST: Task and Data Description

2.1 Task Description

The organisers of EXIST proposed a shared task on automatic detection of multilingual sexist content on Twitter and Gab, including content in English (EN) and Spanish (ES). Two different subtasks were proposed:

- **Subtask 1 - Sexism Identification:** A binary classification task, where every system has to determine whether a given text (tweet or gab) is sexist or not sexist, where sexist content is defined as that which "is sexist itself, describes a sexist situation or criticises a sexist behaviour."
- **Subtask 2 - Sexism Categorisation:** Aiming to classify the sexist texts according to five categories of sexist behaviour including: "ideological and inequality", "stereotype and dominance", "objectification", "sexual violence" and "misogyny and non-sexual violence".

Predictions should be made on a mixed test set including content in both languages. Subtask 1 is evaluated in terms of accuracy, while Subtask 2 is evaluated using a macro-F1 score. Each participating team could submit a maximum of 3 runs.

2.2 Data Description

The EXIST dataset, provided by organisers, consists of 6,977 tweets for training and 3,386 tweets for testing, both of which include content in English and Spanish, and are manually labeled by crowdsourced annotators. In addition, the test set also includes 982 "gabs" from the uncensored social network Gab.com in order to measure the difference between social networks with and without "content control", Twitter and Gab.com respectively. Table 1 shows more details of the datasets provided.

Table 1. EXIST dataset description.

Subtask 1	**Training**		**Testing**		**Subtask 2**	**Training**		**Testing**	
	EN	ES	EN	ES		EN	ES	EN	ES
Sexist	1636	1741	1158	1123	**ideological-inequality**	386	480	333	288
					stereotyping-dominance	366	443	262	257
					sexual-violence	344	401	215	202
					misogyny-non-sexual-violence	284	244	198	202
					objectification	256	173	150	177
Non-sexist	1800	1800	1050	1037	**Non-Sexist**	1800	1800	1050	1037
Total						3436	3541	2208	2160
						6977		4368	
									Twitter: 3386
									Gab: 982

3 The QMUL-SDS System

In this section, we introduce our proposed model XRCNN-Ex and experimental settings. Figure 1 shows the overall framework of the system we submitted to handle the two EXIST subtasks, which uses the pre-trained multilingual model XLM-R with the text-based Convolution Neural Network (TextCNN) and lexical features. We first obtain multilingual semantic information from the hidden state (the last 4 hidden layers) of XLM-R, and then concatenate them together as the input to TextCNN for further feature extraction. External domain knowledge in the lexicon is incorporated into the basic structure of XRCNN and merged with the output of TextCNN. Finally, we pass the merged output features through a dense layer and utilise a softmax function for the final classification.

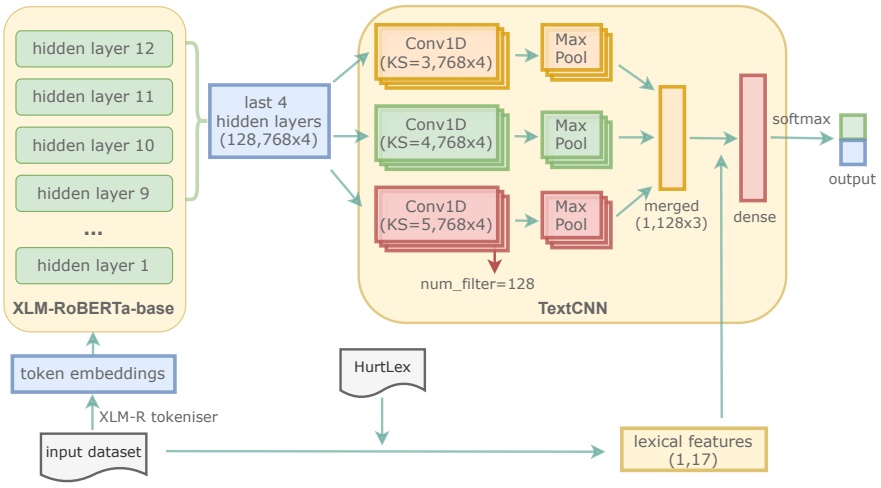

Fig. 1. The overview of XRCNN-Ex architecture.

3.1 XLM-RoBERTa

Previous work with multilingual masked language models (MLM) has proved the effectiveness of pre-training large transformer models on multi-language corpora at once in the domain of cross-lingual understanding [39], such as multilingual BERT (BERT) [9] and cross-lingual language model (XLM) [8]. These models have substantiated their superiority over supervised learning models in many NLP tasks, especially in cases with limited training data. However, both mBERT and XLM are pre-trained on Wikipedia, leading to a relatively limited scale specifically for languages with poor resources. The XLM-RoBERTa model (XLM-R) [7] has extended the way of pre-training MLM by scaling the amount of data by two orders of magnitude (from Wikipedia to Common Crawl) and training on longer sequences (similar to RoBERTa [28]). It has been trained in more than 100 languages, leading to significant improvements on the performance of cross-lingual transfer tasks. In this work, we utilise XLM-R to address the multilingual EXIST dataset and extract semantic features of the whole text to deepen the understanding of the sentence and reduce the impact of noise.

The first token of the sequence in the last hidden layer of XLM-R is commonly used as the output for the classification task, while this output is usually not able to summarise abundant semantic information of the input sentence. Recent work by [21] indicates that richer semantic features can be learned by several hidden layers on top of BERT. In our system, we assume that some top hidden layers of XLM-R are also able to capture semantic information due to the similar architecture of XLM-R and BERT. Thus, we propose the model XRCNN-Ex as shown in Figure 1 for this task. Firstly, the input is processed by the XLM-R tokeniser and fed into the XLM-R model to get a list of hidden states. Then we gain deeper semantic features by integrating the last 4 hidden layers of XLM-R

and feed it into TextCNN. The shape of the output is $n \times (d \times 4)$, where n is the length of the input sentence, and d is the dimension of each token in one hidden layer.

3.2 TextCNN

A text-based Convolutional Neural Network (TextCNN) is a popular architecture for dealing with NLP tasks with a good feature extraction capability [24,43]. The network structure of TextCNN is a variant of the simple CNN model. It is comparatively simpler than other neural networks and is able to reduce the number of dimensions of the input features, resulting in a smaller number of parameters, lower computational needs, and a faster training speed [43]. TextCNN utilises several sliding convolution filters to capture local textual features [24].

In our system, we use multiple 1D convolution kernels at a time for the convolution operation over the output of last 4 hidden states from XLM-R. The output feature set is $X = [x_1, x_2, x_3, ..., x_n] \in \mathbb{R}^{n \times (d \times 4)}$. Let the window $x_{i:i+j-1} = [x_i, x_{i+1}, ..., x_{i+j-1}]$ refer to the concatenation of j words. A filter $w \in \mathbb{R}^{j \times (d \times 4)}$ is involved in the convolution process, applied to the window $x_{i:i+j-1}$ of j words to generate a new feature c_i:

$$c_i = f(w \cdot x_{i:i+j-1} + b) \tag{1}$$

where f is a non-linear function such as ReLU and $b \in \mathbb{R}^{(d \times 4)}$ is the bias. After the filter w slides across $[x_{1:j}, x_{2:j+1}, ..., x_{n-j+1:n}]$, a feature map is generated:

$$C = [c_1, c_2, ...c_{n-j+1}] \in \mathbb{R}^{(n-j+1)} \tag{2}$$

Then we apply the global max-pooling operation over the feature map C and take the maximum value $\hat{c} = max\{C\}$ to capture the most important feature for each feature map [6]. Features extracted by multiple filters are merged and fed into a dense layer.

3.3 Lexical Feature Induction

Currently, language models based on the transformer architecture have been popular among many NLP tasks in both monolingual and multilingual scenarios. But one of the drawbacks is that these models do not take any additional domain knowledge into consideration, like linguistic information from the domain-specific lexicon [42]. Bassignana et al. [3] introduce HurtLex, a multilingual lexicon containing offensive, aggressive, and hateful words and phrases in over 50 languages and spanning 17 categories [3]. The work by Koufakou et al. [25] incorporated lexical features based on the word categories derived from HurtLex to boost the performance of monolingual BERT in such hate-related tasks, whereas there is no relevant study for the multilingual sexism scenario.

Given the scarcity of sexism-specific lexicons as well as the strong relation between those phenomena of offensive language and sexist language [34], we employ HurtLex for the induction of external lexical information to explore how the external lexical features affect the sexism detection performance. We extract 8,228 words for English and 5,006 for Spanish from HurtLex version 1.2, and construct multilingual lexical representations based on the HurtLex categories in both languages. There are 17 diverse categories, described with the number of terms in each language in Table 2. More specifically, we first generate a 17-dimensional lexical vector to count the frequency of each category. For instance, if a text includes 2 words in the category of derogatory words (CDS), the corresponding element of CDS in the lexical vector is supposed to be 2. Then we convert the lexical vector from the count frequency to term frequency–inverse document frequency (TF-IDF) [23], indicating how significant a category is to a text in the corpus. Finally, we concatenate the TF-IDF lexical vector with merged output of the TextCNN, and put it into the dense layer.

Table 2. The category label, description and corresponding number of English and Spanish terms in HurtLex.

Label	**Category Description**	**EN Terms**	**ES Terms**
PS	negative stereotypes ethnic slurs	371	203
RCI	locations and demonyms	24	14
PA	professions and occupations	192	109
DDF	physical disabilities and diversity	63	36
DDP	cognitive disabilities and diversity	491	332
DMC	moral and behavioral defects	715	361
IS	words related to social and economic disadvantage	124	75
OR	plants	177	173
AN	animals	996	679
ASM	male genitalia	426	328
ASF	female genitalia	144	90
PR	words related to prostitution	276	165
OM	words related to homosexuality	361	213
QAS	with potential negative connotations	518	349
CDS	derogatory words	2204	1285
RE	felonies and words related to crime and immoral behavior	619	272
SVP	words related to the seven deadly sins of the Christian tradition	527	322

3.4 Output Layer

In order to prevent the model from over-fitting, we add the dropout after the dense layer, then using a softmax function to obtain the label probability as the final output of the model.

3.5 Experimental Setting

Training Set Split: We use stratified sampling (StratifiedShuffleSplit) in the scikit-learn Python package for the cross-validation step instead of ordinary k-fold cross-validation to evaluate the model. Stratified Shuffle Split is able to create splits by preserving the same percentage for each target class as in the original training set. We set the number of splits to 5 and the ratio of training set to validation set to be 9 to 1. For the EXIST training set, this led to a randomly sampled training set (6,279) and validation set (698). We present all performance scores in Section 4 based on the first split of training and validation sets.

Text Preprocessing: Since texts are obtained from Twitter and Gab, a pre-processing step is needed to maximise the features that can be extracted and to gain a unique and meaningful sequence of words, including removing non-alphabetic words, consecutive white spaces, and lowercasing all texts. As for special tokens in Twitter and Gab, we tokenise hashtags into separate words using the wordsegment Python package, for example: *#HashtagContent* becomes *Hashtag Content*. URLs are replaced with the meta-token <URL> and user names are replaced with <USERNAME>. The text is subsequently tokenised using the corresponding XLM-R pre-trained tokeniser for both languages.

Model Parameter Setting: The parameters in each part of XRCNN-Ex are shown below:

- **XLM-R:** we use XLM-RoBERTa-base pre-trained model, consisting of 12 hidden layers. We set the output hidden states in XLM-R config file to True in order to obtain different hidden states.
- **TextCNN:** we set the number of filters to 128 and three kernel sizes of 3, 4, and 5. ReLU is the non-linear function used for convolution operation.
- **Dense layer:** we set the number of units to 768.

Training Process: During our training process, we use sparse categorical cross entropy as the loss function to save time in memory and computation. We use the Adam optimiser with a learning rate of $1e^{-5}$. We set the max sequence length to 128 and the dropout rate to 0.4. The model is trained in 7 epochs with the batch size of 32. All implementations are under the environment of Keras 2.5.0 and Tensorflow 2.5.0 with python 3.7. The evaluation metrics are accuracy score and macro-averaged f1 score for both two subtasks.

4 Experiments and Results

In this section, we report our results in the two subtasks of the EXIST competition. We first conduct comparative experiments to delve into the optimal way of consolidating features from the hidden state of XLM-R, and then perform an ablation study of the whole architecture of XRCNN-Ex to probe the contribution of its different components. All results are evaluated on the training and validation sets from the first split of original training data released by the EXIST. The official results in the EXIST shared task are presented and discussed finally.

4.1 Comparative Experiments for XLM-R Outputs

The pooler output is commonly utilised as the output of pre-trained language models to address the classification task, which is generally lacking in sufficient and effective semantic information in the sentence representation [21]. More semantic features can be explored from different hidden states of models.

In our experiments, we consider both pooler output and hidden state as the outputs of XLM-R, as well as investigate the consequence of diverse aggregations of several hidden layers. These experiments are implemented on the basic model structure XRCNN and results are displayed in Table 3. It can be observed that integrating the last 4 hidden states of XLM-R yields better performance than other outputs on both subtasks, showing a notable increase in comparison with the pooler output. To be more precise, the model with only the pooler output performs better than the one combining the last 2 hidden layers in subtask 1 and the one with the last hidden layer in subtask 2. Nevertheless, it does not outperform the model absorbed in more than 2 hidden layers, which designates the constraint of the pooler output as the output features and the benefit of abundant semantic information in the hidden layer of XLM-R infused in our model.

Table 3. The XRCNN performance in different aggregations of hidden layers in XLM-R.

XLM-R Hidden Layers	**Subtask 1**		**Subtask 2**	
	Accuracy	Macro F1	Accuracy	Macro F1
Pooler Output	0.754	0.753	0.609	0.527
Last Hidden Layer	0.768	0.768	0.651	0.561
Last 2 Hidden Layers	0.749	0.747	0.645	0.565
Last 3 Hidden Layers	0.801	0.799	0.625	0.541
Last 4 Hidden Layers	**0.804**	**0.804**	**0.663**	**0.590**

4.2 Ablative Experiments and Results

Our proposed model XRCNN-Ex combines the last 4 hidden states of XLM-R and the TextCNN with 3 kernels, then inducting extra lexical information. Several ablative experiments are implemented by removing certain components of XRCNN-Ex to understand the contribution of each component. The following models are applied in this step:

- **XLM-R Last 4 Hidden Layers:** we aggregate the last 4 hidden states of XLM-R as the sentence representations of the input and put them into a simple linear classifier.
- **FastText + TextCNN:** we use the Fasttext embeddings trained on Common Crawl and Wikipedia in 157 languages [18] to convert the input data into word embeddings, and then feed them into a TextCNN.
- **XRCNN:** basic architecture of our proposed model.
- **XRCNN-Ex:** our proposed model incorporating lexical embeddings.

Results of the ablation study are reported in Table 4. We can see that XRCNN and XRCNN-Ex both achieve competitive performance, with noticeable improvements over the other two ablative models XLM-R Last 4 Hidden Layers and FastText+TextCNN. Moreover, XRCNN-Ex achieves a slight improvement in subtask 1 but it does not outperform XRCNN in subtask 2, which casts some doubt on the impact of extra lexical embeddings. We further discuss this in the Section 5.

Table 4. Ablation experiments for different components of XRCNN-Ex.

Model	**Subtask 1**		**Subtask 2**	
	Accuracy	Macro F1	Accuracy	Macro F1
XLM-R Last 4 Hidden Layers	0.788	0.788	0.639	0.539
FastText+TextCNN	0. 751	0.750	0.622	0.528
XRCNN	0.804	0.804	**0.663**	**0.590**
XRCNN-Ex	**0.806**	**0.805**	0.657	0.543

4.3 Official Results in the EXIST Shared Task

Table 5 presents the official results of different runs we submitted to handle the two subtasks as well as the best scores for the EXIST shared task. For these two subtasks, we submitted the results of XRCNN and XRCNN-Ex. The results of XRCNN led to better final scores than XRCNN-Ex, obtaining the better ranks 14^{th} in subtask 1 (accuracy score of 0.761) and 5^{th} in subtask 2 (macro f1 score of 0.559). For the team ranking, we ranked 11^{th} in subtask 1 and 4^{th} in subtask 2.

Table 5. Official results on the test set.

Model	Subtask 1				Subtask 2			
	Accuracy	Macro F1	Rank (runs)	Rank (team)	Accuracy	Macro F1	Rank (runs)	Rank (team)
XRCNN	0.761	0.761	14	11	0.643	0.559	5	4
XRCNN-Ex	0.756	0.756	18	12	0.635	0.546	13	10
Best score	0.780	0.780	-	-	0.659	0.579	-	-

5 Discussion

Our results show that the inclusion of the hidden state of XLM-R and TextCNN effectively improves the model quality of identifying sexist content, which is the most significant contribution of this work. However, results on the test set for XRCNN model with lexical features demonstrate that the choice of lexicon words needs to be done more carefully, as they can lead to harming performance as is the case of XRCNN-Ex in the final scores. We foresee the need to further investigate the following variations to assess their impact on the performance:

- **Dataset variety:** The lexical terms found in the training and test sets might be imbalanced. There may be a certain gap in the quantity of lexical terms extracted in the proportion of the training and test sets, leading to the diverse degree of the influence of lexical terms in the process of the model classification.
- **Term inconsistency between dataset and lexicon:** Terms in the dataset and the lexicon could be inconsistent. The hate-specific lexicon might not be capable of covering all hate-related terms encountered across different datasets.
- **Linguistic characteristics:** Not all posts containing hateful terms are sexist necessarily, due to cases of polysemy or negation.
- **Humour, irony and sarcasm:** Sexist posts with humour, irony and sarcasm are implicit and difficult to be identified, and may contain no explicit hate-related terms.
- **Spelling variation:** Spelling variation is prevalent in social media [40]. Sensitive words sometimes use spelling variations to obfuscate and avoid detection, which do not match those normative words in the lexicon.
- **Quality of lexical features:** TF-IDF frequency features captured from the category of lexical terms might be comparatively sparse and lose information for specific terms. Lexical embeddings derived from pre-trained word embedding models could be beneficial as high-quality word embeddings can be learned efficiently thanks to low space and time complexity [17].
- **Approaches for lexicon induction:** Since the approach for lexicon induction might not fully absorb lexical information by simple concatenation between textual hidden features and lexical features, other forms of fusion can be tested, such as matrix multiplication [35] and cosine similarity [27].

6 Conclusion

In this paper, we describe the participation of the QMUL-SDS team in the EXIST shared task on multilingual sexism identification in English and Spanish social media. As part of our submission, we propose a novel system called XRCNN-Ex. Our submission for binary sexism identification subtask achieves an accuracy score of 0.761 on the test set, ranking 14^{th} among submissions and 11^{th} among teams. For the finer-grained sexism categorisation (subtask 2), we achieve a macro-averaged F1 score of 0.559, ranking 5^{th} and 4^{th} respectively among submissions and teams.

Our basic architecture XRCNN (and XRCNN-Ex), instead of only using the pooler output as the XLM-R's output to deal with the classification task, incorporates the last 4 hidden layers of XLM-R to gain deeper and richer semantic representations, which is fed into a faster classifier TextCNN. Results in both validation and test sets indicate the effectiveness of using multiple hidden states with enriched semantic information and the capability of the TextCNN classifier on top of XLM-R. In addition, we delve into the impact of integrating hate-related lexical embeddings into the system XRCNN-Ex. The results in the validation set show that XRCNN-Ex has a positive influence on subtask 1, while final results in the test set present an inferior performance on both subtasks. We aim to investigate further how to best leverage lexical information.

7 Acknowledgements

Aiqi Jiang is funded by China Scholarship Council (CSC). This research utilised Queen Mary's Apocrita HPC facility, supported by QMUL IT service.

References

1. Anzovino, M., Fersini, E., Rosso, P.: Automatic identification and classification of misogynistic language on twitter. In: International Conference on Applications of Natural Language to Information Systems. pp. 57–64. Springer (2018)
2. Basile, V., Bosco, C., Fersini, E., Nozza, D., Patti, V., Pardo, F.M.R., Rosso, P., Sanguinetti, M.: 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. pp. 54–63 (2019)
3. Bassignana, E., Basile, V., Patti, V.: Hurtlex: A multilingual lexicon of words to hurt. In: 5th Italian Conference on Computational Linguistics, CLiC-it 2018. vol. 2253, pp. 1–6. CEUR-WS (2018)
4. Chiril, P., Moriceau, V., Benamara, F., Mari, A., Origgi, G., Coulomb-Gully, M.: An annotated corpus for sexism detection in French tweets. In: Proceedings of the 12th Language Resources and Evaluation Conference. pp. 1397–1403. European Language Resources Association, Marseille, France (2020)
5. Chiril, P., Moriceau, V., Benamara, F., Mari, A., Origgi, G., Coulomb-Gully, M.: He said "who's gonna take care of your children when you are at acl?": Reported sexist acts are not sexist. In: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. pp. 4055–4066. Online (2020)

6. Collobert, R., Weston, J., Bottou, L., Karlen, M., Kavukcuoglu, K., Kuksa, P.: Natural language processing (almost) from scratch. Journal of Machine Learning Research **12**(76), 2493–2537 (2011)
7. Conneau, A., Khandelwal, K., Goyal, N., Chaudhary, V., Wenzek, G., Guzmán, F., Grave, E., Ott, M., Zettlemoyer, L., Stoyanov, V.: Unsupervised cross-lingual representation learning at scale. In: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. pp. 8440–8451. Association for Computational Linguistics, Online (2020)
8. Conneau, A., Lample, G.: Cross-lingual language model pretraining. In: Advances in Neural Information Processing Systems. vol. 32. Curran Associates, Inc. (2019)
9. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: 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). pp. 4171–4186. Association for Computational Linguistics, Minneapolis, Minnesota (2019)
10. Fersini, E., Nozza, D., Rosso, P.: Overview of the evalita 2018 task on automatic misogyny identification (ami). In: EVALITA@ CLiC-it (2018)
11. Fersini, E., Nozza, D., Rosso, P.: Ami@ evalita2020: Automatic misogyny identification. Proceedings of the 7th evaluation campaign of Natural Language Processing and Speech tools for Italian (EVALITA 2020), Online. CEUR. org (2020)
12. Fersini, E., Rosso, P., Anzovino, M.: Overview of the task on automatic misogyny identification at ibereval 2018. In: IberEval@ SEPLN. pp. 214–228 (2018)
13. Frenda, S., Ghanem, B., Montes-y Gómez, M., Rosso, P.: Online hate speech against women: Automatic identification of misogyny and sexism on twitter. Journal of Intelligent & Fuzzy Systems **36**(5), 4743–4752 (2019)
14. Gagliardone, I., Gal, D., Alves, T., Martinez, G.: Countering online hate speech. Unesco Publishing (2015)
15. Ghosh Chowdhury, A., Sawhney, R., Shah, R.R., Mahata, D.: #YouToo? detection of personal recollections of sexual harassment on social media. In: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. pp. 2527–2537. Association for Computational Linguistics, Florence, Italy (2019). https://doi.org/10.18653/v1/P19-1241, `https://www.aclweb.org/anthology/P19-1241`
16. Glick, P., Fiske, S.T.: Ambivalent sexism. In: Advances in experimental social psychology, vol. 33, pp. 115–188. Elsevier (2001)
17. Goldberg, Y., Levy, O.: word2vec explained: deriving mikolov et al.'s negative-sampling word-embedding method. arXiv preprint arXiv:1402.3722 (2014)
18. Grave, E., Bojanowski, P., Gupta, P., Joulin, A., Mikolov, T.: Learning word vectors for 157 languages. In: Proceedings of the International Conference on Language Resources and Evaluation (LREC 2018) (2018)
19. Hellinger, M., Pauwels, A.: 21. language and sexism. In: Handbook of language and communication: Diversity and change, pp. 651–684. De Gruyter Mouton (2008)
20. Hewitt, S., Tiropanis, T., Bokhove, C.: The problem of identifying misogynist language on twitter (and other online social spaces). In: Proceedings of the 8th ACM Conference on Web Science. pp. 333–335 (2016)
21. Jawahar, G., Sagot, B., Seddah, D.: What does BERT learn about the structure of language? In: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. pp. 3651–3657. Association for Computational Linguistics, Florence, Italy (2019)

22. Jha, A., Mamidi, R.: When does a compliment become sexist? analysis and classification of ambivalent sexism using twitter data. In: Proceedings of the second workshop on NLP and computational social science. pp. 7–16 (2017)
23. Jing, L.P., Huang, H.K., Shi, H.B.: Improved feature selection approach tfidf in text mining. In: Proceedings. International Conference on Machine Learning and Cybernetics. vol. 2, pp. 944–946. IEEE (2002)
24. Kim, Y.: Convolutional neural networks for sentence classification. In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP). pp. 1746–1751. Association for Computational Linguistics, Doha, Qatar (2014)
25. Koufakou, A., Pamungkas, E.W., Basile, V., Patti, V.: HurtBERT: Incorporating lexical features with BERT for the detection of abusive language. In: Proceedings of the Fourth Workshop on Online Abuse and Harms. pp. 34–43. Association for Computational Linguistics (2020)
26. Li, W., Gao, S., Zhou, H., Huang, Z., Zhang, K., Li, W.: The automatic text classification method based on bert and feature union. In: 2019 IEEE 25th International Conference on Parallel and Distributed Systems (ICPADS). pp. 774–777. IEEE (2019)
27. Li, X., Song, J., Liu, W.: Label-attentive hierarchical attention network for text classification. In: Proceedings of the 2020 5th International Conference on Big Data and Computing. pp. 90–96 (2020)
28. Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., Stoyanov, V.: Roberta: A robustly optimized bert pretraining approach. arXiv preprint arXiv:1907.11692 (2019)
29. Manne, K.: Down girl: The logic of misogyny. Oxford University Press (2017)
30. Montes, M., Rosso, P., Gonzalo, J., Aragón, E., Agerri, R., Álvarez-Carmona, M.A., Álvarez-Mellado, E., Carrillo-de Albornoz, J., Chiruzzo, L., Freitas, L., Gómez Adorno, H., Gutiérrez, Y., Jiménez Zafra, S.M., Lima, S., Plaza-de Arco, F.M., Taulé, M.: In: Proceedings of the Iberian Languages Evaluation Forum (IberLEF 2021), CEUR Workshop Proceedings (2021)
31. Mulki, H., Ghanem, B.: The first arabic misogyny identification shared task as a subtrack of hasoc @fire2021. `https://sites.google.com/view/armi2021/` (2021), [Online; accessed 01-06-2021]
32. Nobata, C., Tetreault, J., Thomas, A., Mehdad, Y., Chang, Y.: Abusive language detection in online user content. In: Proceedings of the 25th International Conference on World Wide Web. pp. 145–153. International World Wide Web Conferences Steering Committee (2016)
33. Nozza, D., Volpetti, C., Fersini, E.: Unintended bias in misogyny detection. In: IEEE/WIC/ACM International Conference on Web Intelligence. p. 149–155. WI '19, Association for Computing Machinery, New York, NY, USA (2019). https://doi.org/10.1145/3350546.3352512, `https://doi.org/10.1145/3350546.3352512`
34. Pamungkas, E.W., Basile, V., Patti, V.: Misogyny detection in twitter: a multilingual and cross-domain study. Information Processing & Management **57**(6), 102360 (2020)
35. Pappas, N., Henderson, J.: Gile: A generalized input-label embedding for text classification. Transactions of the Association for Computational Linguistics **7**(0), 139–155 (2019)
36. Richardson-Self, L.: Woman-hating: On misogyny, sexism, and hate speech. Hypatia **33**(2), 256–272 (2018)

37. Rodríguez-Sánchez, F., Carrillo-de Albornoz, J., Plaza, L.: Automatic classification of sexism in social networks: An empirical study on twitter data. IEEE Access **8**, 219563–219576 (2020)
38. Ródriguez-Sánchez, F., de Albornoz, J.C., Plaza, L., Gonzalo, J., Rosso, P., Comet, M., Donoso, T.: Overview of exist 2021: sexism identification in social networks. Procesamiento del Lenguaje Natural **67**(0) (2021)
39. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L.u., Polosukhin, I.: Attention is all you need. In: Advances in Neural Information Processing Systems. vol. 30. Curran Associates, Inc. (2017)
40. Vidgen, B., Harris, A., Nguyen, D., Tromble, R., Hale, S., Margetts, H.: Challenges and frontiers in abusive content detection. In: Proceedings of the Third Workshop on Abusive Language Online. pp. 80–93. Association for Computational Linguistics, Florence, Italy (2019)
41. Waseem, Z., Hovy, D.: Hateful symbols or hateful people? predictive features for hate speech detection on twitter. In: Proceedings of the NAACL student research workshop. pp. 88–93 (2016)
42. Wiegand, M., Ruppenhofer, J., Schmidt, A., Greenberg, C.: Inducing a lexicon of abusive words–a feature-based approach (2018)
43. Zhang, T., You, F.: Research on short text classification based on textcnn. In: Journal of Physics: Conference Series. vol. 1757, p. 012092. IOP Publishing (2021)