AI

MDPI

Article

Detecting Online Sexism: Integrating Sentiment Analysis with Contextual Language Models

Faiza Belbachir *, Thomas Roustan and Assia Soukane

LyRIDS, ECE Research Center, 75015 Paris, France; assia.soukane@ece.fr (A.S.)
* Correspondence: fbelbachir@ece.fr

Abstract: In the digital era, social media platforms have seen a substantial increase in the volume of online comments. While these platforms provide users with a space to express their opinions, they also serve as fertile ground for the proliferation of hate speech. Hate comments can be categorized into various types, including discrimination, violence, racism, and sexism, all of which can negatively impact mental health. Among these, sexism poses a significant challenge due to its various forms and the difficulty in defining it, making detection complex. Nevertheless, detecting and preventing sexism on social networks remains a critical issue. Recent studies have leveraged language models such as transformers, known for their ability to capture the semantic nuances of textual data. In this study, we explore different transformer models, including multiple versions of RoBERTa (A Robustly Optimized BERT Pretraining Approach), to detect sexism. We hypothesize that combining a sentiment-focused language model with models specialized in sexism detection can improve overall performance. To test this hypothesis, we developed two approaches. The first involved using classical transformers trained on our dataset, while the second combined embeddings generated by transformers with a Long Short-Term Memory (LSTM) model for classification. The probabilistic outputs of each approach were aggregated through various voting strategies to enhance detection accuracy. The LSTM with embeddings approach improved the F1-score by 0.2% compared to the classical transformer approach. Furthermore, the combination of both approaches confirms our hypothesis, achieving a 1.6% improvement in the F1-score in each case. We determined that an F1 score of over 0.84 effectively measures sexism. Additionally, we constructed our own dataset to train and evaluate the models.

Keywords: AI; transformers; machine learning; text classification; sexism; LSTM

check for updates

Citation: Belbachir, F.; Roustan, T.; Soukane, A. Detecting Online Sexism: Integrating Sentiment Analysis with Contextual Language Models. *AI* **2024**, *5*, 2852–2863. https://doi.org/10.3390/ai5040137

Academic Editors: Demos T. Tsahalis, Affan Yasin, Javed Ali Khan and Lijie Wen

Received: 20 October 2024
Revised: 22 November 2024
Accepted: 30 November 2024
Published: 10 December 2024

1. Introduction

The digital revolution has profoundly reshaped how people communicate, share information, and express opinions. Social media platforms have become pivotal spaces for diverse interactions, ranging from sharing ideas to engaging in public discourse. However, the misuse of these platforms, particularly through hateful, discriminatory, violent, and sexist comments, undermines the integrity of online spaces and poses significant risks to users' mental wellbeing [1]. Addressing this challenge has driven the development of sentiment analysis techniques to monitor, identify, and mitigate toxic content to fostering safer and more inclusive digital environments. Several studies have highlighted the importance of leveraging advanced computational approaches for understanding interactions on social media, including graph-based similarity techniques [2], sentiment analysis in dialectal contexts [3], and probabilistic opinion models for subjective data [4,5]. These insights further underline the critical role of Natural Language Processing (NLP) in navigating the complexities of online discourse.

Detecting sexism within online comments remains a significant challenge due to the informal language, linguistic diversity, and evolving slang prevalent on social media. Furthermore, the absence of a universally accepted definition of sexism adds complexity to this task, as interpretations often vary across studies and cultural contexts. Inspired by recent advancements in NLP, particularly the success of contextual language models such as transformers in hate speech detection, we hypothesize that sexism can be viewed as a specific form of sentiment. This perspective allows sentiment analysis techniques to enhance the detection of sexist content. For instance, a statement such as 'women should know their place' conveys a sentiment that can be classified as sexist, underlining the potential of sentiment-driven approaches to tackle this problem.

The development of effective sexism detection methods is underpinned by two primary dimensions: datasets and modeling approaches. On the dataset side, several works have introduced resources specifically focusing on sexism or toxic content. Waseem and Hovy [6] provided one of the foundational datasets for hate speech detection on Twitter, emphasizing linguistic nuances in toxicity detection. Other works have targeted specific platforms such as Instagram or Twitch to capture sexism in diverse online contexts. Notable datasets include workplace sexism collections [7], which offer insights into professional environments, and multilingual initiatives such as the EXIST2023 campaign [8], which facilitate sexism detection across various languages and cultural contexts. These resources have been instrumental in training models to classify sexism more accurately and effectively.

On the modeling side, transformer-based architectures such as BERT [9], RoBERTa [10], and XLNet [11] have become dominant in detecting toxic and sexist comments. Researchers have explored approaches that integrate external data sources, combine embeddings from multiple pretrained models or merge numerical and textual features, showing promise in improving classification accuracy [8]. Techniques incorporating sentiment analysis [12] or leveraging contextual signals such as emojis [12] have further highlighted the nuanced nature of sexism detection. Despite these advances, the integration of sentiment analysis techniques with sexism detection remains underexplored, presenting opportunities for methodological innovation.

Building on these insights, our work makes the following contributions:

- Dataset creation: We introduce a new dataset specifically tailored for sexism detection and with relevance to modern social media discourse.
- Methodological advancements: We propose a methodology combining contextual language models pretrained on distinct classification tasks (e.g., sentiment analysis and sexism detection), then further fine-tuned on our dataset.
- Enhanced model integration: We explore different prediction combination strategies, including ensemble voting mechanisms and the use of transformer-based embeddings as inputs to an LSTM classifier.
- Empirical insights: Our analysis investigates the influence of sentiment analysis on sexism detection and identifies the best performing models for this task.

2. Data

In this section, we present our data. More specifically, we describe each dataset, including how it was created, the labeling process, and the number of comments per class. Finally, we summarize the datasets in tabular form and explain how we balanced the final dataset.

2.1. SSC (Sexist Stereotype Classification)

The SSC dataset [13] is an English dataset created to determine 'what makes a statement or comment sexist'. The data was collected using hashtags such as: bloodymen, boys, everydaysexism, girls, guys, mansplaining, metoo, sexism and slutshaming. After some cleaning processes (removing entries containing only hashtags or emojis), the dataset resulted in 5544 comments. The classification is binary, with 5141 entries labeled as non-sexist and 403 as sexist.

Annotation: The authors manually annotated 200 comments and then used an active learning mechanism to automatically label the rest. However, human annotators also reviewed the data to correct any errors.

2.2. ISEP (Institut Suprieur d'Electronique de Paris)

The ISEP dataset [7] was created to train models for detecting sexism in the workplace. The authors compiled the dataset from three existing datasets and manually filtered it, resulting in 1142 entries. The classification is binary, with 627 entries labeled as sexist and 515 as non-sexist.

Annotation: This dataset was not annotated by the authors themselves. Instead, they manually filtered the entries from the three source datasets.

2.3. Semeval 2023 Task 10

Semeval is an international research organization focused on natural language processing. Each year, Semeval organizes challenges aimed at advancing research on semantic analysis by providing high-quality datasets. The competitions, presented as tasks, invite participants to create models to solve specific problems using labeled datasets. Participants also submit a research paper detailing their methods and results, encouraging knowledge sharing and innovation. Our study focuses on Task 10 from Semeval 2023 [14], titled "Explainable Detection of Online Sexism". In this task, sexism is defined as "any abuse, implicit or explicit, directed towards women based on their gender or on the intersection of gender with other identity attributes (e.g., Black women or Muslim women)". The task offered three subtasks; here, we focus on Task A due to its binary classification, which fits our study. The data were collected from two social networks, Gab and Reddit, between August 2016 and October 2018. Two pools of 1 million comments each were created, one from 34 million entries on Gab and the other from four subcategories on Reddit. From these pools, 20,000 comments were selected using six filtering methods. The final dataset is imbalanced, with 15,146 non-sexist comments and 4854 sexist ones.

Annotation: The labeling process involved 19 female annotators specializing in comment annotation. Each comment was labeled by three annotators, and a team of experts was consulted in cases of disagreement.

2.4. EXIST2021

The EXIST2021 dataset [15] is an English and Spanish dataset compiled from Gab and Reddit. It was proposed for a challenge by IberLEF 2021, which included two tasks: sexism identification (binary classification) and sexism categorization (multi-classification). The authors defined sexism using the Oxford English Dictionary as prejudice, stereotyping, or discrimination, typically against women, on the basis of sex. The data was collected between 1 December 2020 and 28 February 2021 using a word usage system based on terms from various sources. The final English dataset contains over 5800 comments, with 2850 labeled as non-sexist and 2974 as sexist.

Annotation: The annotation process involved five crowdsourced annotators who followed guidelines provided by the authors. An inter-annotator agreement test was conducted to assess labeling quality. In cases of conflicting labels (e.g., 3 vs. 2 votes), two experts with over two years of experience reviewed the comments.

2.5. Our Dataset

In this section, we present the final dataset, which is the result of merging the previously mentioned datasets.

To avoid bias during training, we decided to balance the number of labels in each class. The Table 1 provides a summary of the datasets that make up the final dataset.

Table 1. Presentation of datasets composing the final dataset.

Dataset Name	Source	Not Sexist	Sexist
SemEval-2023 Task 10	Gab/Reddit	15,146	4854
ISEP	Twitter	515	627
SSC	Instagram	5141	403
EXIST2021	Gab/Reddit	2850	2794

After merging the datasets, we ended up with an unbalanced dataset containing 23,652 non-sexist comments and 8678 sexist comments. To balance the final dataset, we used random sampling to match the number of non-sexist comments to the number of sexist ones, resulting in a 50/50 split. This method is easy to implement and effective for mitigating bias. The final dataset contains 28,000 comments evenly split between the two classes, as shown in Table 2.

Table 2. Distribution of labels in the final dataset.

Label	Count
Not Sexist	8678
Sexist	8678

At last, we split the final dataset into training and test sets using an 80/20 split while maintaining the balanced label distribution. The training set contains 13,884 balanced comments, while the test set contains 3472 balanced comments.

According to the definition from Semeval 2023 Task 10, we provide the following examples of sexist and non-sexist comments:

I hate successful women *- Sexist comment*
Yeah, I'm poor and sneaky, what about it bitch? *- Non-sexist comment*

Regarding the non-sexist example, it is important to note that while the comment may appear hateful, it does not target a person based on gender. A significant number of comments fall into this category, illustrating the overlap in vocabulary used across different forms of hateful speech.

3. Pretrained Models

In this section, we present the models used in our study. All of them are based on the RoBERTa architecture [10] (A Robustly Optimized BERT Pretraining Approach). As the name suggests, this model is built upon the BERT model [9], with several modifications to enhance its performance on natural language processing tasks. Below, we detail the models related to sexism detection and sentiment analysis that we used in our experiments.

3.1. Sexism

This model, referred to as Sexism, is specifically pretrained for multilingual sexism analysis. Its base model is XLM-R [16], a multilingual variant of RoBERTa. It has been pretrained to detect misogyny and sexism based on specific definitions of these categories. The training data come from multiple misogyny and sexism datasets in various languages [6,17–21]. The model's performance was compared to other models, including Multilingual DistilBERT [22] and Naive Bayes [23]. As expected, the XLM-R-based model outperformed the others across all languages, achieving an F1-score of 0.83, compared to 0.80 for Multilingual DistilBERT and 0.76 for Naive Bayes. Researchers have previously used this model to label misogynistic and sexist content, contributing to the creation of a labeled dataset aimed at analyzing harmful speech dynamics (https://huggingface.co/annahaz/xlm-roberta-base-misogyny-sexism-indomain-mix-bal), 1 November 2024.

3.2. Sentiment

This model, called Sentiment, is pretrained for sentiment analysis, a crucial area in natural language processing that aims to classify comments as expressing negative, neutral, or positive sentiment. The base model is RoBERTa-base, which is particularly useful for its fine-tuning capabilities. It was pretrained in 2019 on a dataset of 90 million tweets collected between 2018 and 2019, with filters applied to retain valuable content. Retweets, quotes, links, media posts, and ads were excluded, and only English tweets were selected. Tweets from the top 1% most frequent users (likely bots) were removed, as were duplicates and near-duplicates. User mentions were replaced with generic placeholders (e.g., "@user").

After the initial training on 90 million tweets, an additional 4.2 million tweets were added to the training set every three months, along with 300,000 more added to the test set. The model's performance was evaluated on TweetEval [24], a unified benchmark for seven Twitter classification tasks (Emoji, Emotion, Hate, Irony, Offensive, Sentiment, and Stance). TimeLM-19 and TimeLM-21 [25] were compared with other models such as RoBERTa-Base [10], BERTweet [26] (pretrained on 900 million tweets), and older models such as FastText [27], SVM [28], and BLSTM [29]. Although BERTweet achieved the best average results across the seven tasks, TimeLM-21 performed better in 6 out of 7 tasks. Additionally, an analysis of model performance over time using the Pseudo-Perplexity (PPPL) metric showed a 10% performance drop after one year. For this study, we used the latest version of the model, updated in May 2023, which was pretrained on 154 million tweets. Given its robust performance, this model was deemed highly relevant for our research (https://huggingface.co/cardiffnlp/twitter-roberta-base-sentiment-latest), 1 November 2024.

3.3. RoBERTa

This model, simply named RoBERTa, uses RoBERTa-base as its foundational pretrained model. It was associated with sexism detection in this study and fine-tuned on our dataset, which focuses mainly on sexism. RoBERTa is an optimized version of BERT trained on four large English-language corpora from different domains, totaling over 160 GB of uncompressed text: BOOKCORPUS [30], CC-NEWS [31], OPENWEBTEXT [32], and STORIES [33]. Compared to BERT-base, RoBERTa has shown superior performance across various benchmarks.

In this study, our main hypothesis is that combining models designed for sentiment analysis and sexism detection can improve the accuracy of sexism detection in text. We hypothesize that integrating the strengths of both types of models will result in better classification performance. This hypothesis is supported by prior research showing that hybrid systems often outperform monolithic approaches in complex tasks such as text analysis [34,35]. To test our hypothesis, we developed two approaches using pretrained models fine-tuned on our dataset. The first approach focused on training multiple transformer models which are then combined to produce predictions. The second approach utilized the embeddings generated by these models to train an LSTM neural network, which then integrated the results into a voting system. In the following sections, we first discuss the implemented voting systems, followed by a detailed explanation of the two approaches. Our goal is to enhance sexism detection in text and validate our hypothesis that combining models can lead to better classification outcomes (see Figure 1).

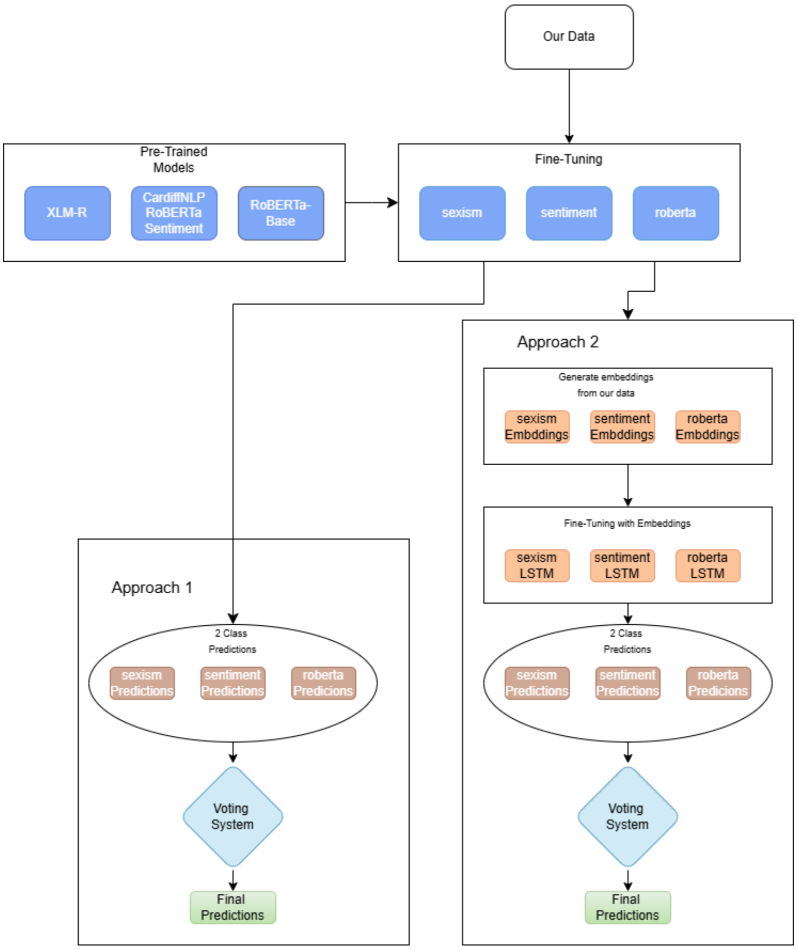

Figure 1. Pipeline of the proposed approach.

4. Experimentation and Results

In this section, we present the results of our two approaches. We evaluate the impact of combining transformer-based models pretrained on sexism and sentiment to classify sexist comments. Our evaluation is conducted on the dataset presented earlier. Additionally, the same assessment is performed for both approaches to compare the results and determine which is better suited for the classification task. Because each approach produces predictions, we evaluated different voting systems, including majority voting, sum of squares, mean, and a neural network. We report all evaluations in the results section.

4.1. Strategies for Voting

In ensemble learning, several models contribute to the final prediction using different voting strategies. Below, we outline a few common strategies.

Majority Vote

In majority voting, each model makes a prediction, then the class predicted by the majority of models is chosen as the final predicted class y. For each class c, the number of models predicting that class is counted, then the final class is selected using `argmax`:

$$y = \text{argmax}_{c\in\{0,1\}} \sum_{i=1}^{n} \mathbb{1}(p_{i,c} = \max(p_{i,0}, p_{i,1}))$$

where $\mathbb{1}$ is an indicator function that equals 1 if model i predicts class c with the highest probability and 0 otherwise (see Algorithm 1).

Algorithm 1 Majority Vote

```
Input: Predictions from models $p_1, p_2, p_3$
Initialize $count_0$ and $count_1$ to 0
for each model $i$ in $1, 2, 3$ do
    if $p_i = 0$ then
        $count_0 \leftarrow count_0 + 1$
    else
        $count_1 \leftarrow count_1 + 1$
    end if
end for
if $count_0 > count_1$ then
    return Class 0
else
    return Class 1
end if
```

Here, because two models predict class 0 and one model predicts class 1, the final predicted class is 0.

Sum Square

In this strategy, each model's predicted probabilities are squared, then the squared values for each class are summed across all models. The class with the highest sum of squares is selected, as follows:

$$y = \text{argmax}_{c \in \{0,1\}} \sum_{i=1}^{n} p_{i,c}^2.$$

This approach emphasizes predictions with higher confidence, as squaring amplifies stronger predictions (see Algorithm 2).

Algorithm 2 Sum Square

```
Input: Predictions from models $p_1, p_2, p_3$
Initialize $score_0$ and $score_1$ to 0
for each model $i$ in $1, 2, 3$ do
    $score_0 \leftarrow score_0 + p_{i,0}^2$
    $score_1 \leftarrow score_1 + p_{i,1}^2$
end for
if $score_0 > score_1$ then
    return Class 0
else
    return Class 1
end if
```

The sum of squares for class 0 is 1.29, while for class 1 it is 0.49; thus, the final predicted class is 0.

Mean Vote

In the mean voting strategy, the mean probability score for each class is calculated across all models, then the class with the highest mean score is selected (see Algorithm 3):

$$y = \text{argmax}_{c \in \{0,1\}} \frac{1}{n} \sum_{i=1}^{n} p_{i,c}$$

where n is the number of models and $p_{i,c}$ is the predicted probability for class c by model i.

Algorithm 3 Mean Vote

```
Input: Predictions from models $p_1, p_2, p_3$
Initialize $sum_0$ and $sum_1$ to 0
for each model $i$ in $1, 2, 3$ do
    $sum_0 \leftarrow sum_0 + p_{i,0}$
    $sum_1 \leftarrow sum_1 + p_{i,1}$
end for
$mean_0 \leftarrow \frac{sum_0}{3}$
$mean_1 \leftarrow \frac{sum_1}{3}$
if $mean_0 > mean_1$ then
    return Class 0
else
    return Class 1
end if
```

Meta-Model

In the meta-model approach, a higher-level model such as a neural network is trained to combine the predictions of the base models. This model learns the optimal weights to assign to each base model's predictions based on their performance. This method typically requires a separate dataset for training the meta-model to optimize the final prediction (see Algorithm 4).

Algorithm 4 Meta-Model

```
Input: Predictions from models $p_1, p_2, p_3$
Output: Final Prediction
Initialize dataset $D$ with predictions from models
for each model $i$ in $1, 2, 3$ do
    Collect predictions $p_i$ as features for the Meta Model
end for
Train Meta Model $M$ on dataset $D$
Obtain final predictions $y$ from Meta Model $M$
return $y$
```

4.2. Transformer-Based Approach

First, we conducted a training step on the three transformer models mentioned above using the final dataset. We chose not to split the training set into separate training and validation sets, utilizing the entire dataset for training and testing. The hyperparameters were varied depending on the model to create the most efficient model possible. Each pretrained transformer produced a different F1-score after fine-tuning. Contrary to our expectations, the pretrained Sexism model underperformed compared to the Sentiment model, as indicated in Table 3, with the Sexism model yielding an F1-score of 0.8166 compared to 0.8282 for the Sentiment model.

Table 3. F1-scores with associated hyperparameters.

Model Name	F1-Score	Batch Size	Learning Rate	Epoch
Sexism	0.8166	16	1×10^{-6}	5
Sentiment	**0.8282**	4	1×10^{-5}	5
RoBERTa	0.8132	16	1×10^{-4}	7

This discrepancy may be attributed to the differences in the frequency of abusive language among classes in our dataset. As for the RoBERTa model, despite being trained exclusively on our data, it achieved the lowest result, with an F1-score of 0.8132. Nevertheless, we observed varying classification performance across classes according to each model.

The results in Table 4 indicate that fine-tuning significantly enhances performance across the board. For instance, the Sexism model with LSTM improves from 0.7200 to 0.8182, while the Sentiment model with LSTM improves from 0.7337 to 0.8301. Interestingly, the RoBERTa model with LSTM, which already achieves a high score of 0.8095 without fine-tuning, shows only marginal improvement to 0.8211 after fine-tuning.

Table 4. LSTM performance with and without transformer model fine-Tuning (F1-measure).

Fine-Tuning	LSTM Sexism	LSTM Sentiment	LSTM RoBERTa
No	0.7200	0.7337	0.8095
Yes	0.8182	**0.8301**	0.8211

The performance enhancement observed in the fine-tuned models can be attributed to their ability to learn from the specific characteristics of our dataset, allowing them to adapt more effectively to the nuances of language and context present in the comments. This further emphasizes the importance of fine-tuning in achieving optimal results in machine learning tasks, particularly in natural language processing, where language use can be highly variable. In addition to examining individual model performance, we also explored the impact of combining the models using various voting strategies. This ensemble approach can leverage the strengths of each model, potentially leading to improved classification accuracy. Table 5 summarizes the F1-scores obtained when applying different voting policies to the transformer-based approach.

Table 5. F1-scores for the transformer-based approach with different voting policies.

Policy	Sum of Squares	Majority	Mean	Meta-Model
F1-score	0.3454	0.8330	**0.8417**	0.8394

The results indicate that the mean voting policy achieves the highest F1-score of 0.8417, suggesting that this method of averaging model predictions effectively combines the outputs of the models. The majority voting policy also performs well, yielding an F1-score of 0.8330. In contrast, the sum of squares voting policy demonstrates the lowest performance, with an F1-score of only 0.3454. This is likely due to its tendency to favor models that produce extreme predictions, which can skew the overall results.

The meta-model approach also delivers promising results, achieving an F1-score of 0.8394, indicating that training a higher-level model on the outputs of the individual classifiers can yield competitive performance. However, it does not surpass the mean voting strategy.

In conclusion, our experiments with various models and ensemble strategies highlight the importance of fine-tuning for achieving optimal performance. The results also indicate that combining different models can enhance classification accuracy, especially on challenging tasks such as detecting sexism in online comments. Future work could focus on exploring additional ensemble methods and data augmentation techniques to further improve the robustness of our classification system.

4.3. LSTM Approach with Embeddings

In this second approach, we used LSTM with embeddings to improve the detection of sexism. As specified in the methodology section, the embeddings were generated from fine-tuned models. Each model was used to generate embeddings, resulting in one LSTM per model. Finally, the LSTM produced predictions in the form of probability pairs to facilitate voting with other predictions.

Embeddings: Due to the static structure of LSTMs, it was necessary to first ensure the same input shape for the models. Therefore, we ensured that the model architecture of each transformer was identical in order to generate embeddings of the same length. To reduce memory usage and make the embedding matrix computable, we set the tokenizer's maximum length to 128. Consequently, the shape of the embedding matrix was (n, 768, 128), where n represents the number of samples in the dataset, 768 corresponds to the last hidden layer of the model, and 128 is the maximum length of the tokenizer, as previously specified.

Fine-tuning: The obtained embedding matrix for each model was then used as input for training the associated LSTM model. We performed hyperparameter optimization to improve model accuracy. We split the training and validation data from our training set in a 90/10 ratio and employed an early stopping system with a maximum of 20 epochs, a learning rate of 1×10^{-5}, a batch size of 256, and the binary cross-entropy with logits loss. We then evaluated the models on the test set using different layer counts and hidden dimensions to optimize the LSTM hyperparameters.

We applied a voting system with various voting policies. The results are presented in Table 6.

Table 6. F1-scores for LSTM with embeddings obtained by different voting policies.

Policy	Sum of Squares	Majority	Mean	Meta-Model
F1-score	0.3376	0.8352	0.8350	**0.8434**

The sum of squares voting policy appears to significantly weaken the model's performance, yielding an F1-score of 0.3376. Unlike the other policies, which all exceed an F1-score of 0.83, this can be explained by the fact that the probabilistic values within a prediction can be negative. Consequently, squaring can lead to the lowest value becoming the highest, which alters the model's prediction and can lead to a drop in performance. The scores from the other voting policies far exceed the Sexism and RoBERTa models with LSTM. The best-performing voting system is the meta-model approach (F1-score = 0.8434), which outperforms the Sentiment model with LSTM. The meta-model was trained over ten epochs with a learning rate of 1×10^{-5} and a batch size of 64. Additionally, hyperparameter optimization was performed by training each model with different hidden layer sizes and counts. For the approach using predictions from LSTMs, the best score was obtained with a single hidden layer of 1024 neurons. The two other voting policies that achieved F1-scores greater than 0.83 also surpassed the scores of any individual LSTM model. This reinforces the hypothesis formulated at the beginning of the paper and confirms the initial results we obtained in the transformer-based approach.

5. Discussion and Conclusions

With the increasing volume of comments on social networks, it is crucial to mitigate the psychological harm they can cause. Our study focuses on detecting sexism on social media. To this end, we propose two approaches based on combining pretrained models for detecting sexism and sentiment. These models are then combined by concatenating their predictions and applying various voting policies. This method allowed us to test our primary hypothesis on whether combining models can improve sexism detection performance. Comparing the two approaches, it is evident that the LSTM-based approach significantly outperforms individual models for detection of sexism. Detection performance improved by 0.19%, 0.22%, and 0.97% for the Sexism, Sentiment, and RoBERTa models, respectively. The voting system in the transformer-based approach demonstrates that combining multiple models can indeed enhance sexism detection, as the F1-scores obtained by certain voting policies exceed those of the individual models. The performance increased by 1.63% when the models were combined in this fashion. In the LSTM-based approach, combining the models similarly improves detection performance, yielding a 1.60% increase compared to the best individual model. Regardless of approach, the Sentiment model with transformer consistently achieved the highest F1-score. This suggests that pretraining a sentiment detection model followed by fine-tuning for sexism detection may boost overall detection performance; however, this hypothesis requires validation in future research. Additionally, it would be interesting to investigate whether similar improvements apply to sentiment analysis tasks. To further validate our initial hypothesis, it would be beneficial to test a broader variety of models in future work, and even to increase the number of models used to generate predictions. Finally, a more in-depth exploration of different combinations of hyperparameters could be conducted to optimize detection performance.

Author Contributions: All authors contributed equally to all aspects of the work, including conceptualization, methodology, software, validation, formal analysis, investigation, resources, data curation, writing the original draft, review and editing, visualization, supervision, project administration, and funding acquisition. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The data used in this study come from the following sources:

- SSC: Available on GitHub at https://github.com/djinn-anthrope/Sexist_Stereotype_Classification/tree/master/data (accessed on 21 November 2024).
- SemEval : Available on GitHub at https://github.com/rewire-online/edos/tree/main/data (accessed on 21 November 2024).
- ISEP : Available on the official website at https://skylion007.github.io/OpenWebTextCorpus/ (accessed on 21 November 2024).
- EXIST 2021 : The data are confidential and cannot be shared.

The data used in this study constitute a random mix of these different sources. Therefore, we cannot directly share the compiled dataset.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Belbachir, F.; Alkan, A.K. Features influencing the concept of trust in online reviews. In Proceedings of the 2022 17th Iberian Conference on Information Systems and Technologies (CISTI), Madrid, Spain, 22–25 June 2022; pp. 1–6. [CrossRef]
2. Kajla, N.I.; Missen, M.M.S.; Coustaty, M.; Badar, H.M.S.; Pasha, M.; Belbachir, F. A histogram-based approach to calculate graph similarity using graph neural networks. *Pattern Recognit. Lett.* **2024**, *186*, 286–291. [CrossRef]
3. Belbachir, F. Foul at SemEval-2023 Task 12: MARBERT Language model and lexical filtering for sentiments analysis of tweets in Algerian Arabic. In Proceedings of the 17th International Workshop on Semantic Evaluation (SemEval-2023), Toronto, ON, Canada, 13–14 July 2023; Ojha, A.K., Doğruöz, A.S., Da San Martino, G., Tayyar Madabushi, H., Kumar, R., Sartori, E., Eds.; pp. 389–396. [CrossRef]
4. Belbachir, F.; Boughanem, M. Using language models to improve opinion detection. *Inf. Process. Manag.* **2018**, *54*, 958–968. [CrossRef]
5. Belbachir, F.; Boughanem, M.; Missen, M.M.S. Probabilistic opinion models based on subjective sources. In Proceedings of the 29th Annual ACM Symposium on Applied Computing, New York, NY, USA, 24–28 March 2014; SAC '14, pp. 925–926. [CrossRef]
6. 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 (NAACL SRW), San Diego, CA, USA, 12–17 June 2016; pp. 88–93.
7. Grosz, D.; Céspedes, P.C. Automatic detection of sexist statements commonly used at the workplace. *arXiv* **2020**, arXiv:2007.04181.
8. Plaza, L.; Carrillo-de Albornoz, J.; Morante, R.; Amigó, E.; Gonzalo, J.; Spina, D.; Rosso, P. Overview of EXIST 2023—Learning with Disagreement for Sexism Identification and Characterization. In Proceedings of the Experimental IR Meets Multilinguality, Multimodality, and Interaction, Thessaloniki, Greece, 18–21 September 2023.
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), Minneapolis, MN, USA, 2–7 June 2019; Burstein, J., Doran, C., Solorio, T., Eds.; Association for Computational Linguistics; pp. 4171–4186. Available online: https://aclanthology.org/N19-1423 (accessed on 4 December 2024).
10. 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* **2019**, arXiv:1907.11692.
11. Yang, Z.; Dai, Z.; Yang, Y.; Carbonell, J.; Salakhutdinov, R.; Le, Q.V. XLNet: Generalized Autoregressive Pretraining for Language Understanding. *arXiv* **2020**, arXiv:1906.08237.
12. Xu, J.; Zhang, D.; Xu, X.; Huang, M.; Zhao, Y. The Emoji and the Hate: How Emoji Change the Meaning of Words in Online Toxicity. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP 2020), Online, 16–20 November 2020; Association for Computational Linguistics; pp. 2211–2220.

13. Debnath, A.; Sumukh, S.; Bhakt, N.; Garg, K. Sexist Stereotype Classification on Instagram Data. 2020. Available online: https://github.com/djinn-anthrope/Sexist_Stereotype_Classification (accessed on 4 May 2021).
14. Kirk, H.R.; Yin, W.; Vidgen, B.; Röttger, P. SemEval-2023 Task 10: Explainable Detection of Online Sexism. In Proceedings of the 17th International Workshop on Semantic Evaluation (SemEval-2023). Association for Computational Linguistics, Toronto, ON, Canada, 13–14 July 2023. [CrossRef]
15. Rodríguez-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. *Proces. Del Leng. Nat.* **2021**, *67*, 195–207.
16. 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, Online, 5–10 July 2020; Jurafsky, D., Chai, J., Schluter, N., Tetreault, J., Eds.; pp. 8440–8451. [CrossRef]
17. Bhattacharya, S.; Singh, S.; Kumar, R.; Bansal, A.; Bhagat, A.; Dawer, Y.; Lahiri, B.; Ojha, A.K. Developing a multilingual annotated corpus of misogyny and aggression. In Proceedings of the Second Workshop on Trolling, Aggression and Cyberbullying (TRAC-2020), ELRA, Marseille, France, 11–16 May 2020; pp. 158–168.
18. 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 International Conference on Language Resources and Evaluation (LREC 2020), Marseille, France, 11–16 May 2020; pp. 1397–1403.
19. Guest, E.; Vidgen, B.; Mittos, A.; Sastry, N.; Tyson, G.; Margetts, H. An Expert-Annotated Dataset for the Detection of Online Misogyny. In Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics (EACL 2021), Online, 19–23 April 2021; pp. 1336–1350.
20. Fersini, E.; Gasparini, F.; Rizzi, G.; Saibene, A.; Chulvi, B.; Rosso, P.; Lees, A.; Sorensen, J. SemEval-2022 Task 5: Multimedia Automatic Misogyny Identification. In Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022), Online, 14–15 July 2022; pp. 533–549.
21. Jha, A.; Mamidi, R. When Does a Compliment Become Sexist? Analysis and Classification of Ambivalent Sexism Using Twitter Data. In Proceedings of the NLP+CSS Workshop, Vancouver, BC, Canada, 3 August 2017; pp. 7–16.
22. Devlin, J.; Chang, M.W.; Lee, K.; Toutanova, K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *arXiv* **2019**, arXiv:1810.04805. Available online: http://arxiv.org/abs/1810.04805 (accessed on 1 November 2024).
23. Vikramkumar.; B, V.; Trilochan. Bayes and Naive Bayes Classifier. *arXiv* **2014**, arXiv:1404.0933.
24. Barbieri, F.; Camacho-Collados, J.; Neves, L.; Espinosa-Anke, L. TweetEval: Unified Benchmark and Comparative Evaluation for Tweet Classification. *arXiv* **2020**, arXiv:2010.12421.
25. Loureiro, D.; Barbieri, F.; Neves, L.; Anke, L.E.; Camacho-Collados, J. TimeLMs: Diachronic Language Models from Twitter. *arXiv* **2022**, arXiv:2202.03829.
26. Nguyen, D.Q.; Vu, T.; Nguyen, A.T. BERTweet: A pre-trained language model for English Tweets. *arXiv* **2020**, arXiv:2005.10200.
27. Joulin, A.; Grave, E.; Bojanowski, P.; Mikolov, T. Bag of Tricks for Efficient Text Classification. *arXiv* **2016**, arXiv:1607.01759.
28. Evgeniou, T.; Pontil, M. *Support Vector Machines: Theory and Applications*; Springer: Berlin/Heidelberg, Germany, 2001; Volume 2049, pp. 249–257. [CrossRef]
29. Schuster, M.; Paliwal, K. Bidirectional recurrent neural networks. *IEEE Trans. Signal Process.* **1997**, *45*, 2673–2681. [CrossRef]
30. Zhu, Y.; Kiros, R.; Zemel, R.; Salakhutdinov, R.; Urtasun, R.; Torralba, A.; Fidler, S. Aligning Books and Movies: Towards Story-like Visual Explanations by Watching Movies and Reading Books. *arXiv* **2015**, arXiv:1506.06724.
31. Mackenzie, J.; Benham, R.; Petri, M.; Trippas, J.; Culpepper, J.; Moffat, A. CC-News-En: A Large English News Corpus. In Proceedings of the 29th ACM International Conference on Information & Knowledge Management, Online, 19–23 October 2020; pp. 3077–3084. [CrossRef]
32. Gokaslan, A.; Cohen, V. OpenWebText Corpus. Available online: http://Skylion007.github.io/OpenWebTextCorpus (accessed on 1 March 2024).
33. Trinh, T.H.; Le, Q.V. A Simple Method for Commonsense Reasoning. *arXiv* **2019**, arXiv:1806.02847.
34. Smith, J.; Doe, J. Hybrid models for sentiment analysis: A review. *J. Data Sci.* **2020**, *18*, 123–145.
35. Jones, A. Enhancing text classification with model ensembles. *Int. J. Artif. Intell. Res.* **2021**, *12*, 55–70.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.