Binary classification of sexist sentences

Ester Molinari

April 3, 2025

Abstract

Binary text classification can be performed by various techniques, from machine learning to deep learning ones. In this report, we will train different state-of-the-art models to classify sexist and non-sexist sentences from EDOS Dataset provided for Task 10 of SemEval 2023.

1 Introduction

Detecting sexism in the the written and spoken language is not an easy job. Task 10 of SemEval 2023 proposed the **Explainable Detection of Online Sexism (EDOS) dataset** along with three main tasks to solve, from binary to multiclass classification, in order to detect and explain sexism [2].

In this report, we will focus only on Task A which covers the **binary classification of sexism in sentences** addressing the challenges presented by the dataset itself, leveraging different **machine learning** and **deep learning techniques** for binary classification, such as Support Vector Machines, Classification Trees, and Neural Networks.

2 Related Work

From SemEval final report [2] emerges that, at the end of the competition, **90%** of partecipants used a transformer-based model based on BERT architecture while **8%** of partecipants preferred traditional machine learning methods. Only the remaining percentage of partecipants used non-transformer deep neural networks, which have been combined with other methods.

It has been proven that **fine-tuning RoBERTa models** shows better performances with respect to fine-tuned BERT models on the original dataset [3]. Other partecipants pointed out that the number of sexist data was comparatively low, hinting to the presence of **unbalanced data** [4].

The approaches presented in this report are based on a system that incorporates information related to emotions, polarity, and irony in the texts as **metadata** to get better results [5]. With their experiments they were able to prove that, in general, the incorporation of extra-linguistic information helps the models to conduct the tasks.

3 Proposed approach

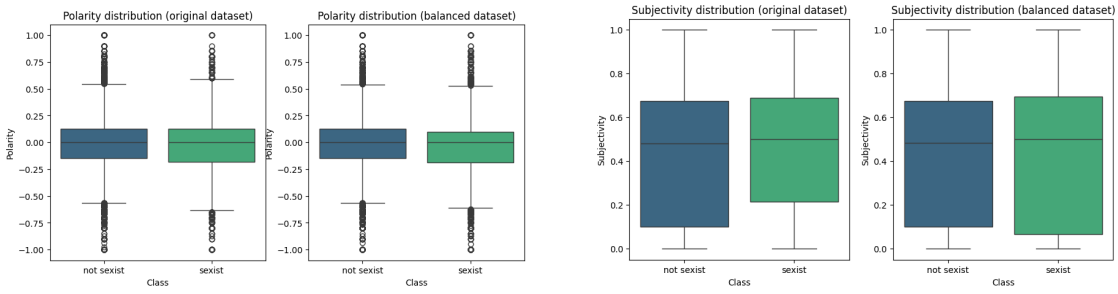
The proposed approaches are performed on two version of the EDOS dataset which are the **original** one and a **balanced** version [6].

3.1 Data exploration and preprocessing

By performing some data analysis and exploration on both dataset versions, it is exposed how the original dataset is **unbalanced** compared to the other one.

Another interesting result is drawn by **polarity** and **subjectivity** distribution in sentences for both datasets, shown in Figure 1. Apparently, polarity and subjectivity cannot be used as features as

they do not allow us to distinguish sexist sentences from non-sexist sentences.



(a) (b)

Figure 1: Polarity and subjectivity of both datasets

Overfitting a Support Vector Machine (SVM) with a linear kernel, we are able to realize if a dataset is linearly separable or not [1]. For this purpose, we trained two SVMs with linear kernel on both datasets to find out that the original dataset shows **non-linear separable data** while the balanced dataset shows **linear separable data**. Due to this information, the original dataset will be used only with **non-linear models** while the balanced one will be used with **linear models**.

In order to choose which models to train, we perform a **10-fold cross-validation** with a default SVM, logistic regression and a perceptron, aiming to the model that is able to score the highest **accuracy**, which is the metric in which we are interested. Choosing a higher recall with respect to precision for a sentence classification task would not be meaningful because of the nature of the data, for this task is better to aim to a good balance between them. The results of the cross-validation are the following:

- Mean accuracy for SVM: **0.81 with a standard deviation of 0.01**
- Mean accuracy for logistic regression: 0.78 with a standard deviation of 0.01
- Mean accuracy for perceptron: 0.71 with a standard deviation of 0.03

Due to this results, we will only train two SVM with different kernels on both datasets. Firstly, we perform a **text preprocessing pipeline** that involves stopword removal and lemmatization, then we get the **embedding** version of each sentence in order to train every model. From now on, when we are talking about sentences, we are referring to **sentence embeddings** obtained by BERT sentence transformer.The results will be shown in the related section.

3.2 Support Vector Machines for binary classification

The hyperparameters for both SVMs are obtained performing a **randomized search with a 3-fold cross-validation** aiming to the highest accuracy. The selected hyperparameters are the following:

1. SVM on original data with **RBF kernel**, class weight **1:5** for sexist sentences and C at **100**
2. SVM on balanced data with **linear kernel**, class weight **balanced** and C at **1**

3.3 Classification Trees for binary classification

Random Forest and **Gradient Boosting trees** have been trained only on the balanced dataset, as they are suitable for binary classification. The hyperparameters for both models are obtained performing, again, a **randomized search with a 3-fold cross-validation** aiming to the highest accuracy. The selected hyperparameters are the following:

1. Random Forest with **300 estimators**, minimum samples split at **2**, minimum samples leaf at **4**, maximum depth at **30** and **no class weight**

2. Histogram Gradient Boosting with minimum samples leaf at **10**, maximum iteration at **300**, **no maximum depth**, learning rate set to **0.2** and L1 regularization at **1**

3.4 Neural network for sentences embeddings binary classification

The architecture of this neural network can be described as follows. It is a **feedforward neural network with two hidden layers** (with 128 and 64 neurons respectively) and a single output neuron with a **sigmoid activation function** that has been trained for 100 epochs with an early stopping that triggered at the 39th epoch. The learning rate is set to 0.0001 and the weight decay at 0.0001.

3.5 Neural network for ngrams embedding binary classification

By conducting experiments based on the **cosine similarity** between the embeddings of word ngrams, interesting results emerged. It seems that there exists "sexist trigrams" and "non-sexist trigrams" that can be used to classify sentences performing a **majority voting technique**.

The experiment consisted in extracting the **trigrams** of sexist sentences and the trigrams of non-sexist sentences, obtaining **two distinct sets**. After turning all the trigrams into embeddings, it becomes possible to perform cosine similarity between the trigrams and see if the chosen trigram is more sexist or more non-sexist. To classify a sentence, a **majority voting technique** is then applied classifying the sentence as sexist or not based on the number of sexist trigrams detected. This simple experiment shows interesting results in Table 1, like the following for the sentence *She's asking for it dressed like that.*

Original trigram	Most similar sexist trigram	Most similar non-sexist trigram	Classification
She's asking for	woman want actually	ask want woman	Non-sexist
asking for it	people ask want	ask person ask	Sexist
for it dressed	outfit need wear	f*** need costume	Sexist
it dressed like	outfit look like	dress look like	Sexist
dressed like that.	outfit look like	dress look like	Sexist

Table 1: Trigram classification with cosine similarity from a sexist sentence

Following these results, we trained a **feedforward neural network with two hidden layers** (with 128 and 64 neurons respectively) and a single output neuron with a **sigmoid activation function** which is able to classify a sentence starting from its trigrams' embeddings. The model has been trained for 50 epochs. The learning rate is set to 0.0001 and the weight decay at 0.0001.

According to the results of the ngrams experiment, the neural network has been trained using the **mean** and **max pooling** of the trigrams embeddings of the sentences, leading to different results.

4 Results

The overall results of every model is shown in Figure 2. At a glance, the simple neural network for sentences embeddings binary classification, named *Simple NN*, seems to outperform every other model, leaving on the lowest step of the podium the SVM with non-linear kernel.

The performance gap between the SVM with non-linear kernel, named *SVM (original)* and the SVM with linear kernel, named *SVM (balanced)* is quite noticeable for every metric.

The two classification trees show very good results, but their strength is due to the time of training. If both SVMs took approximately **25 minutes** to train, the Random Forest took **17 minutes** while the Histogram Gradient Boosting took only **2 minutes**, which is the fastest machine learning model, so far.

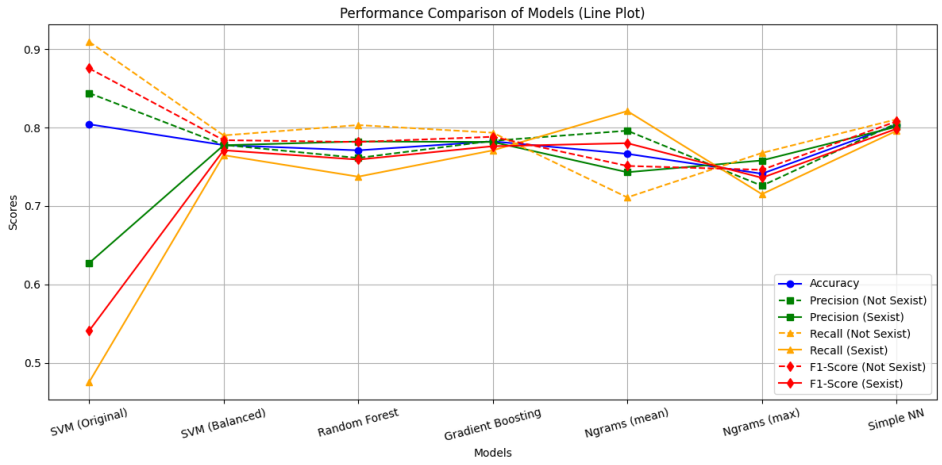


Figure 2: Overall performances of all models

The most interesting results are given by the neural networks. The first neural network, named *Ngrams (mean)* is trained to classify the mean of all the embeddings obtained from the trigrams of sentences while *Ngrams (max)* is trained to classify the maximum value of all the embeddings obtained from the trigrams of sentences. Ngrams (mean) shows better accuracy, recall and F1-Score on sexist sentences with respect to Ngrams (max), as shown in the figure.

The second neural network, which is again **Simple NN**, shows a really nice improvement with respect to every other model, showing the best performances and compromise between the two classes.

5 Conclusions

This report shows the challenges that can arise during the binary classification of a dataset made up of sentences. It was possible to explore different types of state-of-the-art models suitable for the classification task, also playing with lower-level elements, such as ngrams, combining them with high-level techniques from deep learning, such as sentence embeddings.

References

[1] D. Elizondo. The linear separability problem: Some testing methods. *IEEE Transactions on neural networks*, 17(2):330–344, 2006.

[2] H. Kirk, W. Yin, B. Vidgen, and P. Röttger. SemEval-2023 task 10: Explainable detection of online sexism. In A. K. Ojha, A. S. Doğruöz, G. Da San Martino, H. Tayyar Madabushi, R. Kumar, and E. Sartori, editors, *Proceedings of the 17th International Workshop on Semantic Evaluation (SemEval-2023)*, pages 2193–2210, Toronto, Canada, July 2023. Association for Computational Linguistics.

[3] M. Padmavathi. Ds at semeval-2023 task 10: Explaining online sexism using transformer based approach. In *Proceedings of the 17th International Workshop on Semantic Evaluation (SemEval-2023)*, pages 1102–1106, 2023.

[4] R. H. Rifat, A. Shruti, M. Kamal, and F. Sadeque. Acsmkrhr at semeval-2023 task 10: Explainable online sexism detection (edos). In *Proceedings of the 17th International Workshop on Semantic Evaluation (SemEval-2023)*, pages 724–732, 2023.

[5] M. E. V. Rodrguez, F. M. P. Del Arco, L. A. U. Lopez, and M. T. Martín-Valdivia. Sinai at semeval-2023 task 10: Leveraging emotions, sentiments, and irony knowledge for explainable detection of online sexism. In *Proceedings of the 17th International Workshop on Semantic Evaluation (SemEval-2023)*, pages 986–994, 2023.

[6] A. Rydelek, D. Dementieva, and G. Groh. AdamR at SemEval-2023 task 10: Solving the class imbalance problem in sexism detection with ensemble learning. In A. K. Ojha, A. S. Doğruöz, G. Da San Martino, H. Tayyar Madabushi, R. Kumar, and E. Sartori, editors, *Proceedings of the 17th International Workshop on Semantic Evaluation (SemEval-2023)*, pages 1371–1381, Toronto, Canada, July 2023. Association for Computational Linguistics.