NLP Assignment 1

Habib Kazemi, Hesam Sheikh Hassani, Ehsan Ramezani
Master's Degree in Artificial Intelligence, University of Bologna
{ habib.kazemi2, hesam.sheikhhassani, ehsan.ramezani }@studio.unibo.it

Abstract

This report covers our approach to tweet sexism detection using a Bidirectional LSTM-based baseline model, an enhanced LSTM architecture, and a fine-tuned transformer model. The dataset was preprocessed using text-cleaning methods tailored to standardize inputs, given the unique challenges of tweet datasets. The experiments show the impact of hyperparameter optimization and various architectures on model performance, with the fine-tuned transformer achieving the highest F1 score.

1 Introduction

Sexism detection is a crucial task in social media to combat online abuse. The task comes down to classifying texts into sexist and non-sexist categories. In this report, we discuss how using transformer networks can enhance the task of sexism detection of tweets. We first explore LSTM-based models as our baseline and then fine-tune a transformer and observe the difference between the two approaches.

In the **System description** section, we map out the LSTM-based models we trained as a baseline, their architectures, the transformer model, and go through the data preprocessing. We point out the unique challenge in this dataset, and explain the preprocessing methods we incorporated to ensure the data is rightly prepared.

The details of the training pipeline of the LSTM and the fine-tuning methods and hyperparameters of the transformer model are explained in the **Experimental setup and results section**, in which we also discuss the results of our evaluations.

2 System description

We use the EXIST 2023 Task 1 (1) dataset, which consists of 2870 labeled tweets for training. It is crucial to standardize the tweets as much as possible to reduce the number of tokens not present in the **GloVe** (2) embeddings (we use 'Wikipedia 2014 + Gigaword 5' version). This would make more tokens meaningful in the embedding space, thus helping the training and fine-tuning stage later on. The casual nature of tweets make this particularly difficult. We leverage a collection of methods to remove emojis (3), hashtags, mentions, URLs, etc. We use two additional methods not suggested by the assignment, which are to **lowercase** all tweets and **correct typos**; helping us to reduce the number of tokens not found in Glove from 4469 to 1194.

We train three variations of Bidirectional LSTM models: **BaselineModel** (527 K parameters) is our most basic, which includes one layer of Bidirectional LSTM coupled with a fully connected layer in the end, **Model 1** (910 K parameters) which adds another LSTM layer to the BaseLineModel, and **Model 2** (704 K parameters) is similar to Model 1, but the hidden dimension between the two LSTM layers is equal. For the transformer model, **Twitter-roBERTa-base** (4) is selected, which is trained on 58M tweets and is 125M parameters. The model is fine-tuned using the transformers(5) library and similar to our LSTM training approach, we fine-tune the transformer on three instances.

3 Experimental setup and results

In training the LSTM models, at the end of each step, the loss, f1 score, and the accuracy of the models are monitored. To ensure the best of the models is selected as the baseline, we train them in three instances, and the Optuna library is utilized to systematically search for the optimal hyperparameter configuration for each model, as shown in Table 1. To save computation while handling variable-length sequences, we packed the sequences. Each model is trained for 60 steps with early stopping, and the results are in Table 2 .

After exploring LSTM architectures, We fine-tuned the `cardiffnlp/twitter-roberta-base-hate` Transformer model from HuggingFace for our task.

Table 1: Hyperparameters for LSTM Models

Hyperparameter	Baseline	Model 1	Model 2
Hidden Dim 1	87	77	86
Hidden Dim 2	-	159.0	-
Dropout Rate 1	0.238	0.267	0.512
Learning Rate	0.0091	0.0059	0.0050
Weight Decay	0.0019	0.0029	0.0027

Table 2: Validation F1-Score and Standard Deviation for Each LSTM Model

Hyperparameter	Baseline	Model 1	Model 2 (M2)
Average Validation F1-Score	0.8316	0.8293	0.8280
Standard Deviation	0.0094	0.0043	0.0051

Using the HuggingFace `Trainer`, , the model was fine-tuned with a learning rate of 2×10^{-5},a batch size of 16, and a linear scheduler with a 0.1 warmup ratio. Three different random seeds (42, 144, and 256) were used for robustness. Early stopping (patience=2, threshold=0.01) was employed based on the validation macro F1-score. The highest F1-score checkpoints were saved. Results are shown in Table 3.

Table 3: Validation F1-Score and Standard Deviation for Transformer Model: `cardiffnlp/twitter-roberta-base-hate`

Model	Seed	Validation F1-Score
`cardiffnlp/twitter-roberta-base-hate`	42	0.8642
`cardiffnlp/twitter-roberta-base-hate`	144	0.8562
`cardiffnlp/twitter-roberta-base-hate`	256	0.8693
Average	–	0.8633
Standard Deviation	–	0.0054

4 Discussion

4.1 Quantitative Results

The Transformer model outperformed the LSTM models on the validation set, achieving a higher average macro F1-score (0.8633 compared to 0.8316 for the best LSTM). This superior performance is attributed to the Transformer's pre-training on a large dataset and its ability to better understand context and long-range relationships in the text.

The LSTM models showed signs of overfitting Table 4, and increasing complexity by adding layers (Model 1 and Model 2) did not improve results, even slightly decreasing the average F1-score. The σ of F1-scores across seeds was slightly lower for Model 1 and Model 2 than the Baseline, indicating increased stability from added layers, but at the cost of a slightly lower mean F1. The Transformer model also showed good stability across different seeds, with a σ of 0.0054.

4.2 Error Analysis

Despite good overall performance, the models still made errors, as seen in misclassifications and confusion matrices. Common error patterns were found, highlighting the difficulties the models faced.

- **Ambiguous Language and Context**
- **Indirect or Complex Sexism**
- **Data Labeling Challenges**
- **Figurative Language and Idioms**
- **Informal and Fragmented Tweets**

Examples:

- "get twitter harass for pro choice" – Illustrates ambiguous language affecting classification.
- "raise taxis and control" – Demonstrates how figurative language can confuse models.

For future development, we suggest the following improvements: enriching the training data with diverse examples of subtle sexism, figurative language, and contextual information; enhancing contextual understanding in LSTM models using attention mechanisms or sliding windows; and incorporating external knowledge sources like lexicons of sexist terms.

5 Conclusion

This study investigated LSTM and Transformer models for sexism detection in tweets. The fine-tuned cardiffnlp/twitter-roberta-base-hate Transformer outperformed the LSTM models, achieving an average macro F1-score of 0.8633, demonstrating the advantage of leveraging pre-trained models. However, all models struggled with nuanced language, indirect sexism, and figurative expressions. Key limitations include the ambiguity of language in tweets and the simplicity of binary classification. Future work should focus on enriching training data, incorporating external knowledge, exploring ensemble methods, and improving preprocessing techniques to address these challenges and enhance model robustness.

6 Links to external resources

The LSTM models weights: link
The transformer model weights: link

A Appendix

This section contains supplementary materials, including additional figures.

References

CLEF 2023 | Conference and Labs of the Evaluation Forum. (n.d.). https://clef2023.clef-initiative.eu/index.php?page=Pages/labs.htmlEXIST

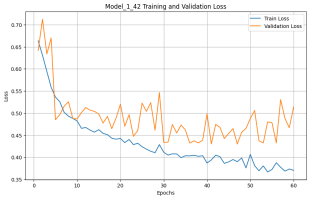
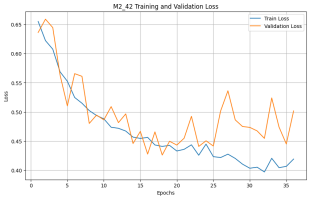
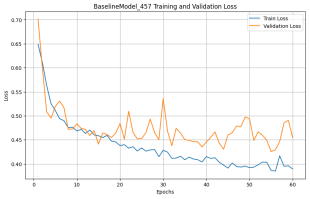
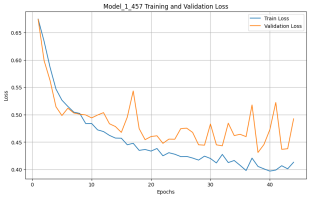
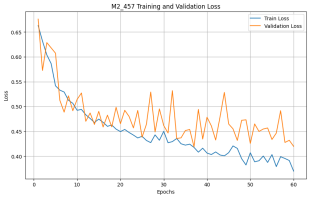
Stanfordnlp. (n.d.). GitHub - stanfordnlp/GloVe: Software in C and data files for the popular GloVe model for distributed word representations, a.k.a. word vectors or embeddings. GitHub. https://github.com/stanfordnlp/GloVe?tab=readme-ov-file

removing emojis from a string in Python. (n.d.). Stack Overflow. https://stackoverflow.com/a/33417311/10187497

cardiffnlp/twitter-roberta-base-hate · Hugging Face. (n.d.). https://huggingface.co/cardiffnlp/twitter-roberta-base-hate

Transformers. (n.d.). https://huggingface.co/docs/transformers/en/index

Table 4: Learning curve figures grouped by models and seeds.

Seed	Baseline	Model 1	Model 2
42			
457			
999