Offensive Speech Classification Study Report by Nerry Asyncrite Koukoui

Abstract

This research paper addresses the detection and removal of offensive language within datasets of user-generated content sourced from various online platforms such as social media, online forums, and comment sections of news websites.The paper highlights previous research that concentrated on identifying particular types of offensive content, such as hate speech, cyberbullying, and cyber aggression, as well as the rising prevalence of offensive content on social media.

The publicly accessible Offensive Language Identification Dataset (OLID) was used to detect offensive language in tweets. The paper discusses various techniques for identifying and categorizing offensive speech and stresses the significance of such initiatives in upholding a secure and welcoming online environment.

1 Materials

• Code

• Google Drive Folder containing models and saved outputs

• Presentation

2 Model Selection (Task 1)

2.1 Summary of 2 selected Models

In this study, the Stochastic Gradient Descent(SGD) Classifier and Recurrent Neural Networks (RNN) were selected for the task.

2.1.1 Stochastic Gradient Descent

The SGDClassifier is a machine learning algorithm that can be used for multi-class classification tasks. It has been demonstrated to achieve state-of-the-art performance on tasks like text classification and sentiment analysis, making it particularly useful for applications in Natural Language Processing (Kabir et al., 2015).The algorithm operates by minimizing a loss function that determines the difference between the actual class labels in the training data and the predicted class labels.

2.1.1 Recurrent neural networks

Recurrent neural networks (RNNs) are a kind of neural network that are effective at handling sequential data. RNNs have a recurrent connection, which enables them to maintain a state or memory of prior inputs, in contrast to feedforward neural networks, which process input data in a single pass (Dadvar et al., 2013c). As a result, they are particularly beneficial for processes like time-series prediction, speech recognition, and natural language processing.

Long Short-Term Memory (LSTM) networks are a popular type of RNN that uses gating mechanisms to selectively retain or forget information from the previous hidden state. This enables the network to not only capture long-term dependencies and retain information over longer time periods but also to excel at tasks involving time series predictions and natural language processing.

2.2 Critical discussion and justification of model selection

The SDG Classifier is a popular algorithm for applications where explainability and interpretability are crucial because it is also a straightforward algorithm that is simple to implement and understand (Linardatos et al., 2020). The SGD classifier is efficient when handling large datasets. These features of the model, especially its track record of success in natural language processing applications (Chen et al., 2020), with high dimensionality, do so effectively through the use of mini-batches which prevents overfitting and improves the model's ability to generalize on unseen data, making it a suitable choice for offensive speech classification (Gaydhani, 2018).

It is important to remember that the SDG Classifier might not be the best option for all applications, especially those that call for the model to recognize

intricate patterns or connections in the data.

Due to their capacity to model sequential data and capture long-term dependencies, recurrent neural networks (RNNs) are a well-liked option for classifying offensive speech. In tasks involving natural language processing, where the context of a sentence frequently determines its meaning, this is especially helpful (Anis, 2017). The capacity of RNNs to handle input sequences of variable length, which is critical in tasks like text classification, is one of their main advantages (Proceedings, 2019). RNNs are able to keep track of data about the entire sequence and use it to generate predictions because they employ a hidden state that is updated with each new input.

In tasks like sentiment analysis, where the meaning of a sentence can be influenced by minute details, RNNs have been shown to effectively capture complex patterns and relationships in data. However, it is important to note that RNNs can be computationally expensive and may need significant computational resources to train effectively. Additionally, they are prone to problems such asexploding and vanishing gradients, which can make training challenging.

The quantity and caliber of the training data may also have an impact on how effective RNNs are. The model might not be able to uncover the underlying patterns in the data if the dataset is too small or lacks sufficient representative examples. Because they can handle sequential data and capture long-term dependencies, RNNs continue to be a popular option for classifying offensive speech. RNNs can perform well if the unique requirements of the task at hand are carefully considered, and if the right techniques, like regularization and early stopping, are used.

3 Design and implementation of Classifiers

1. Learning Rate: In SDG, the size of the step taken in the direction of the gradient during training is determined by the learning rate. A high learning rate can cause the model to diverge, while a low learning rate can result in a slow convergence (Brownlee, 2019).
2. Dropout Rate: RNNs are prone to overfitting due to their sequential nature. Dropout was used to regularize the model and prevent overfitting.
3. Batch Size: The batch size determines the number of samples processed together in each iteration during training. A larger batch size can lead to faster training times but can also result in a slower convergence and a lower quality of the final model.
4. Number of Epochs: the number of times the model is trained on the entire training dataset was determined by the number of epochs.
5. Hidden Layer Size: The size of the hidden layers in the RNN determines the number of neurons used in each layer. The size of the hidden layers can have a significant impact on the capacity and complexity of the model.
6. Loss Function: The choice of loss function determines how the model is optimized during training. For binary classification, you can use Binary Cross-Entropy Loss.
7. Optimizer: The optimizer determined the algorithm used to update the weights of the model during training.

Dataset	Total	%OFF	%NOT
Train	12313	33.2	66.8
Valid	927	33.24	65.75
Test	826	29.06	75.06

Table 1: Dataset Details

Table 1 details the OLID dataset utilized for offensive speech classification. It contains three subsets of data: Train, Valid, and Test, with the corresponding number of tweets in each subset. The table provides insight into the percentage of offensive and non-offensive tweets in each subset. The Train subset contains the most tweets, with 33.2% being offensive and 66.8% being non-offensive. The Valid and Test subsets contain fewer tweets, with 33.24% and 29.06% offensive tweets, respectively. The Test subset has the highest percentage of non-offensive tweets at 75.06%. This information is essential for

understanding the distribution of offensive and non-offensive tweets in the dataset and evaluating the performance of models trained on this data.

Model	F1-score		SoA
	NOT	OFF	
Model 1 (25%)	81	52	73
Model 1 (50%)	81	42	72
Model 1 (75%)	76	46	67
Model 1(100%)	76	46	67
Model 2 (25%)	84	0	42
Model 2 (50%)	84	0	72
Model 2 (75%)	84	0	72
Model 2(100%)	84	0	72

Table 2: Model Performance

From Table 2 it is observed that the F1-scores of Model 1 decrease as the size of the data increases. At 25% data size, Model 1 achieved the highest F1-score of 81% for all three classification tasks. However, as the size of the data increased, the F1-score of Model 1 decreased to 76% for both 75% and 100% data sizes. This suggests that Model 1 overfits the smaller datasets and could not generalize well to larger datasets.

On the other hand, the F1-scores of Model 2 remained constant at 84% for all four data sizes. This indicates that Model 2 is more robust and can generalize well across different data sizes. However, it should be noted that Model 2 did not perform well on the NOT and OFF classification tasks, achieving 0 F1-scores for both.

In conclusion, based on the F1-scores, Model 2 outperforms Model 1 in terms of robustness and generalization across different data sizes. However, Model 1 performed better on the smaller datasets, suggesting overfitting on those datasets. The effect of the different data sizes on the models is that it can affect their generalization and performance, as shown by the decreasing F1-scores of Model 1 as the size of the data increases. Therefore, it is important to choose an appropriate amount of data for training a model to ensure that it can generalize well to unseen data.

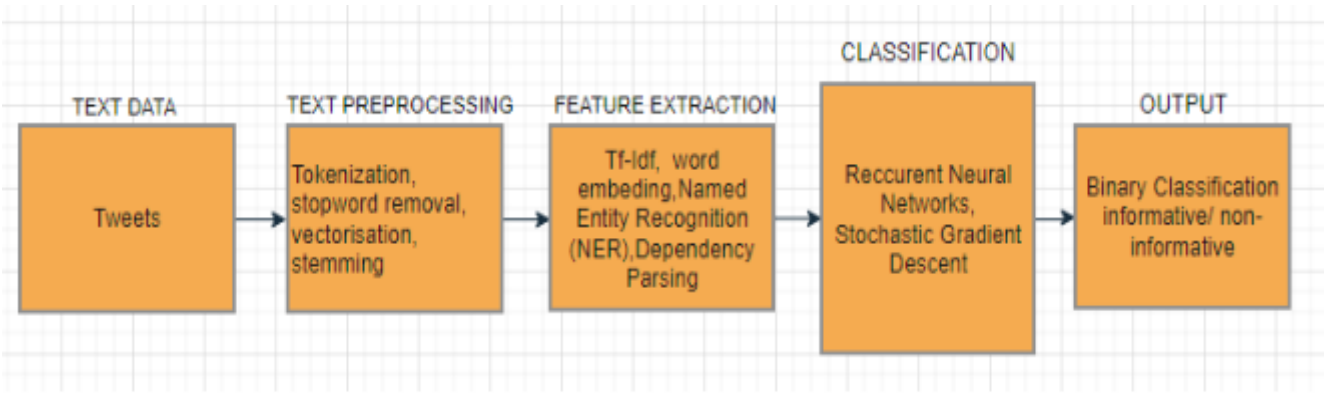

Figure 1: Diagram illustrating pipeline and models

FINDINDS ANS CONCLUSION

This report presents a summary of two selected

models, the Stochastic Gradient Descent (SGD) Classifier and Recurrent Neural Networks (RNN), for the task of offensive speech classification. The SGD classifier is a machine learning algorithm that can be used for multi-class classification tasks, with a track record of success in natural language processing applications. On the other hand, RNNs are effective at handling sequential data with a recurrent connection that enables them to maintain a state or memory of prior inputs. Long Short-Term Memory (LSTM) networks are a popular type of RNN that uses gating mechanisms to selectively retain or forget information from the previous hidden state.

The report also provides a critical discussion and justification of model selection. The SGDClassifier is a popular algorithm for applications where explainability and interpretability are crucial, and it is efficient in handling large datasets, with a track record of success in natural language processing applications. Recurrent neural networks (RNNs) are a well-liked option for classifying offensive speech because of their capacity to model sequential data and capture long-term dependencies, which is helpful in tasks involving natural language processing, where the context of a sentence frequently determines its meaning.

Furthermore, the report provides details of the design and implementation of classifiers, such as learning rate, dropout rate, batch size, number of epochs, hidden layer size, loss function, and optimizer. These parameters determine the effectiveness of the model in training and classification. Finally, the section presents the dataset details utilized for offensive speech classification, containing three subsets of data: Train, Valid, and Test, with the corresponding number of instances and the percentage of offensive and non-offensive speech.

References

[1]Gaydhani, A. (2018b, September 23). *Detecting Hate Speech and Offensive Language on Twitter using Machine Learning: An N-gram and TFIDF based Approach*. arXiv.org. https://arxiv.org/abs/1809.08651

[2] Kwok, I., & Wang, Y. (2013). Locate the Hate: Detecting Tweets against Blacks. *Proceedings of the . . . AAAI Conference on Artificial Intelligence*, *27*(1), 1621–1622. https://doi.org/10.1609/aaai.v27i1.8539

[3] Dadvar, M., Trieschnigg, D., Ordelman, R., & De Jong, F. (2013d). Improving Cyberbullying Detection with User Context. In *Lecture Notes in Computer Science* (pp. 693–696). Springer Science+Business Media. https://doi.org/10.1007/978-3-642-36973-5_62

[4] Haidar, B., Chamoun, M., & Serhrouchni, A. (2017). A Multilingual System for Cyberbullying Detection: Arabic Content Detection using Machine Learning. *Advances in Science, Technology and Engineering Systems Journal*, *2*(6), 275–284. https://doi.org/10.25046/aj020634

[5] Davidson, T., Warmsley, D., Macy, M. W., & Weber, I. (2017). Automated Hate Speech Detection and the Problem of Offensive Language. *National Conference on Artificial Intelligence*, *11*(1), 512–515. https://doi.org/10.1609/icwsm.v11i1.14955

[6] Yin, W., Kann, K., Yu, M., & Schütze, H. (2017). Comparative Study of CNN and RNN for Natural Language Processing. *ResearchGate*. https://www.researchgate.net/publication/313443663_Comparative_Study_of_CNN_and_RNN_for_Natural_Language_Processing

[7] Miro-Llinares, F., & Rodríguez-Sala, J. J. (2016). Cyber hate speech on twitter: Analyzing disruptive events from social media to build a violent communication and hate speech taxonomy. *International Journal of Design & Nature and Ecodynamics*, *11*(3), 406–415. https://doi.org/10.2495/dne-v11-n3-406-415

[8] Anis, M. Y. (2017b). Hatespeech in Arabic Language. *ResearchGate*. https://www.researchgate.net/publication/319553062_Hatespeech_in_Arabic_Language

[9] Gelashvili, T. (2018). *Hate Speech on Social Media: Implications of private regulation and governance gaps*. LUP Student Papers. https://lup.lub.lu.se/student-papers/search/publication/8952399

[10] Fortuna, P., & Nunes, S. (2018). A Survey on Automatic Detection of Hate Speech in Text. *ACM Computing Surveys*, *51*(4), 1–30. https://doi.org/10.1145/3232676

[11] Khurana, D., Koli, A. C., Khatter, K., & Singh, S. (2017). Natural language processing: state of the art, current trends and challenges. *Multimedia Tools and Applications*. https://doi.org/10.1007/s11042-022-13428-4

[12] Linardatos, P., Papastefanopoulos, V., & Kotsiantis, S. (2020b). Explainable AI: A Review of Machine Learning Interpretability Methods. *Entropy*, *23*(1), 18. https://doi.org/10.3390/e23010018

[13] Balci, S., Demirci, G., Demirhan, H., & Sarp, S. (2022). Sentiment Analysis Using State of the Art Machine Learning Techniques. In *Lecture notes in networks and systems* (pp. 34–42). Springer International Publishing. https://doi.org/10.1007/978-3-031-11432-8_3