Deliverable 3, OffensEval - Offensive language identification, SemEval-2019 Task 6

Anusha Goulla, Vyoma Harshitha Podapati, Shree Pallavi Vegesana, Armin Keshavarz Rahbar, Lakshmi Durga Teratipally

George Mason University

Abstract

This study focuses on offensive language identification on social media using the Offensive Language Identification Dataset (OLID) from SemEval-2019 Task 6. OLID provides a hierarchical annotation schema with three levels: classifying tweets as offensive or not, categorizing offensive tweets as targeted or untargeted, and identifying targets of targeted offensive tweets as individuals, groups, or others. We applied preprocessing techniques such as replacing URLs and mentions with placeholders and removing unnecessary special characters, along with tokenization and TF-IDF vectorization. Logistic Regression models were trained for each subtask and evaluated using accuracy, precision, recall, and F1 score. The results showed moderate performance in distinguishing offensive from non-offensive tweets, with a macro F1 score of 0.7348. For distinguishing targeted offensive content from untargeted, the macro F1 score was 0.6530, while identifying the target of the offense achieved a lower macro F1 score of 0.5934, reflecting the difficulty of this multi-class classification task.

Introduction

There is no doubt that social media plays a major role in our lives today, shaping the way we acquire knowledge, interact with others, and share opinions on matters important to us. However, social media also presents a downside, primarily due to the prevalence of offensive language. Without face-to-face interactions, individuals often cross the conventional boundaries of respectful conduct observed during in-person engagements. The ease with which people use offensive language on social media creates an unwelcoming environment that can alienate users and limit healthy discourse. Yet, moderating such content requires walking a fine line between identifying and removing offensive posts and protecting people's freedom of expression. It is therefore important to classify the nature of language in social media posts with accuracy to promote respectful online discourse while upholding user rights.

In our Introduction to Natural Language Processing course, we have explored various foundational aspects of NLP, including essential text preprocessing tasks, such as tokenization, stop word and special characters removal, and punctuation filtering. We have also delved into classification tasks, where we trained machine learning models to classify text based on content. Additionally, we have examined evaluation metrics such as precision, recall, F1 score, and accuracy, which are essential for assessing the performance of NLP models.

Using the skills we have acquired so far; we will analyze Twitter posts to identify and categorize offensive language. We will rely on the Offensive Language Identification Dataset (OLID), which provides a collection of annotated tweets that reflects three levels of offensive language classification. Each level is represented as a separate subtask. Sub-task A involves classifying tweets as either Offensive (OFF) or Not Offensive (NOT). Sub-task B further classifies Offensive

tweets as either Targeted Insults/Threats (TIN) or Untargeted (UNT). Finally, Sub-task C classifies the target of targeted offensive tweets as either directed at an Individual (IND), Group (GRP), or Other (OTH).

We will preprocess the text data by performing several steps, including tokenization, removing stop words, punctuation, and special characters to prepare the data for the training phase. After completing the preprocessing phase, we will train separate classification models tailored for each sub-task, using the respective training data to optimize each model for its specific classification goal. Once the models are trained, we will test and evaluate their performance on corresponding test sets, measuring their effectiveness through metrics such as precision, recall, F1 score, and accuracy.

Related Work

Research on abusive and offensive language covers various areas, including hate speech, aggression, cyberbullying, and toxic content. Early work in these areas established foundational methods using annotated datasets and supervised classification techniques.

Some tasks have provided datasets specifically for identifying aggression, categorizing posts as non-aggressive, covertly aggressive, or overtly aggressive. Others specializing in toxic comment classification have explored datasets with multiple labels, including categories for varying levels of toxicity and identity-based hate. Furthermore, other efforts have focused on offensive language in specific languages, introducing tasks to classify such content further into subcategories like profanity, insults, and abuse.

One of the earlier works done in the area of offensive language detection was the **GermEval Shared Task on the Identification of Offensive Language** (Wiegand et al., 2018), which focused on German tweets. It featured two tasks: a binary classification for identifying offensive tweets and a multi-class classification for more detailed subcategories. These subcategories included profanity, insults, and abuse. Profanity involved the use of swear words or curses without necessarily targeting a specific individual. Insults specifically targeted individuals by giving them an identity, which while social in nature is perceived by the majority of society as degrading.

While these initiatives addressed specific aspects of offensive content, they did not look into the target of the offensive language.

Building on these earlier efforts, the work **Predicting the Type and Target of Offensive Posts in Social Media (**Zampieri et al, 2019**)** introduced a dataset with a hierarchical structure for offensive language detection that aims to also identify the target of the offensive language. The hierarchical dataset deployed in this work enables classification across three levels: detecting the presence of offensive language, categorizing its type (whether it is targeted offensive language such as insult or threat or untargeted, such as general profanity) and identifying its target such as an individual, group or other (any targeted insult that is neither again an individual or group. This approach integrates and expands upon prior methodologies, offering a comprehensive resource for analyzing offensive language.

Data and Methods

Dataset Description:

The Offensive Language Identification Dataset (OLID) is a benchmark dataset widely used for identifying and categorizing offensive language in social media posts. Developed for the OffensEval-2019 shared task in SemEval-2019 Task 6, OLID has become an essential resource in academic research and education, with applications

in universities like the University of Arizona, Imperial College London, and the University of Leeds. The dataset comprises 14,100 English tweets collected from Twitter, divided into 13,240 tweets for training and 860 for testing. It employs a hierarchical three-level annotation schema.

At Level A, tweets are classified as either Offensive (OFF) or Not Offensive (NOT). At Level B, offensive tweets are further categorized as Targeted Insults (TIN) or Untargeted (UNT). Finally, at Level C, targeted insults are assigned one of three targets: Individual (IND), Group (GRP), or Other (OTH), such as organizations or events.

The dataset's hierarchical structure ensures a logical flow, where tweets must first be classified as offensive before being categorized by type or target. This allows for detailed and layered analysis of offensive content. The class distribution within OLID highlights a significant class imbalance, particularly at Levels B and C. For instance, while Level A contains 9,460 non-offensive tweets compared to 4,640 offensive ones, Level B shows only 551 Untargeted tweets compared to 4,089 Targeted Insults. At Level C, Individual (IND) targets dominate with 2,507 tweets, compared to 1,152 targeting Groups (GRP) and only 430 classified as Other (OTH).

The class distribution for OLID is as follows:

Level	Class	Training Instances	Test Instances	Total Instances
Level A	Not Offensive (NOT)	8,840	620	9,460
	Offensive (OFF)	4,400	240	4,640
Level B	Untargeted (UNT)	524	27	551
	Targeted Insult (TIN)	3,876	213	4,089
Level C	Individual (IND)	2,407	100	2,507
	Group (GRP)	1,074	78	1,152
	Other (OTH)	395	35	430

While the dataset provides a strong foundation for offensive language detection, it does have limitations. The class imbalance poses challenges for machine learning models, and the dataset's focus on Twitter-specific content limits its applicability to other platforms like Facebook or Instagram. Additionally, offensive language often relies on subtle context, making it challenging for both annotators and algorithms to capture all nuances accurately.

Despite these challenges, OLID has proven invaluable for advancing research in automated content moderation, hate speech detection, and offensive language categorization. It supports the development of robust machine learning and deep learning models while providing a benchmark for evaluating their performance. By addressing limitations such as class imbalance and extending its methodology to other platforms and languages, OLID continues to contribute to the development of safer and more inclusive online environments.

Methods:

The task of offensive language detection is a multi-level classification problem, divided into three hierarchical subtasks. Sub-task A involves classifying tweets as either offensive (OFF) or not offensive (NOT). Sub-task B focuses on further categorizing offensive tweets as targeted insults or threats (TIN) or untargeted content (UNT). Finally,

Sub-task C identifies the target of targeted insults or threats as an individual (IND), group (GRP), or other (OTH), such as organizations or events. The primary objective is to develop and evaluate classification models for each subtask using robust machine learning techniques.

To prepare the dataset for modeling, several preprocessing steps were applied. First, tweets with missing labels for specific subtasks were removed to ensure a clean and consistent dataset. NaN values were dropped for tasks where annotations were unavailable. The dataset was then split into training (80%) and testing (20%) subsets for each subtask, using stratified splitting to maintain the proportional distribution of class labels and address the dataset's inherent class imbalance.

In this code, we performed comprehensive preprocessing on the tweet data to clean and standardize it for modeling. The preprocessing began by converting all text to lowercase to eliminate discrepancies caused by case sensitivity. Hashtags were processed to remove the # symbol while retaining the associated words to ensure meaningful terms were preserved. URLs present in the text were replaced with the placeholder <URL>, allowing their presence to be acknowledged without contributing unnecessary noise. Similarly, mentions of users, such as @username, were replaced with the placeholder <MENTION>, generalizing user references while maintaining contextual relevance.

Non-alphanumeric characters, except for ! and ?, were removed to clean the text and reduce noise, as these symbols might indicate sentiment or emphasis in the context of tweets. Numeric values were replaced with the placeholder <NUM> to represent the presence of numbers without including specific values, which are typically less informative in text classification tasks.

Finally, extra spaces were removed, and leading or trailing spaces were stripped to ensure a consistent and tidy format. This preprocessing pipeline transformed raw tweet data into a structured and meaningful format suitable for feature extraction and machine learning models.

The cleaned text data was transformed into numerical format using the Term Frequency-Inverse Document Frequency (TF-IDF) method. TF-IDF effectively captures the relative importance of words by emphasizing terms that occur frequently in a document but are rare across the corpus. This method helps reduce the impact of common stop words while highlighting unique and meaningful words. To ensure computational efficiency while retaining meaningful text features, the TF-IDF vectorizer configurations were tailored for each subtask based on experimentation. For Sub-task A and Sub-task B, the vectorizer was set to include a maximum of 10,000 features and incorporated unigrams and bigrams (ngram_range=(1, 2)), which provided improved results by capturing both individual words and word pairs. Additionally, stop_words='english' was applied to remove common stopwords, further enhancing performance. For Sub-task C, the vectorizer performed best with a maximum of 3,000 features and also used unigrams and bigrams. However, applying stop_words='english' in Sub-task C negatively impacted results, so it was excluded. These configurations were carefully adjusted to balance computational efficiency with the need to capture the most relevant and contextual features for each subtask.

For each subtask, classification models were developed using machine learning techniques, like Logistic Regression. Each model was fine-tuned through hyperparameter optimization to enhance performance. Evaluation of the model was conducted using metrics such as accuracy,

precision, recall, and F1-score, with confusion matrices analyzed to understand classification errors. The combination of robust preprocessing, efficient text vectorization, and well-optimized machine learning models ensured the accurate detection and categorization of offensive language in social media posts. This methodology demonstrates the effectiveness of integrating preprocessing techniques and machine learning for addressing complex hierarchical classification problems.

In the hyperparameter optimization phase, we employed GridSearchCV with 5-fold cross-validation to systematically tune the Logistic Regression model's regularization strength. For each subtask, we investigated different ranges of regularization strengths (parameter C): Sub-task A explored values between 0.01 and 10, Sub-task B examined values from 0.0099 to 10, and Sub-task C investigated values ranging from 0.01 to 10. This nuanced approach allowed us to optimize model performance across diverse classification challenges.

This approach enabled a robust selection of the most effective regularization strength by evaluating the model's performance across multiple subsets of the training data, ultimately enhancing the model's generalizability and predictive accuracy for identifying offensive tweets, their type, and target. To address potential class imbalance in the tweet classification task, we configured the Logistic Regression with balanced class weights and set a maximum of 1000 iterations to ensure convergence. The optimization process focused on the F1 Macro score, a metric particularly suitable for multi-class classification.

Model Selection:

Logistic Regression was selected as our primary classification algorithm for detecting offensive language due to its unique capabilities in handling complex text classification tasks. The model provides an optimal balance between computational efficiency and predictive performance, making it particularly suited to processing and analyzing tweet-based data.

The theoretical foundations of Logistic Regression are rooted in its ability to create a probabilistic linear decision boundary that separates different classes in high-dimensional feature spaces. By transforming input features into class probability predictions using the logistic (sigmoid) function, the model effectively translates complex textual data into meaningful classification outcomes. Our model selection was driven by several key advantages of Logistic Regression. The algorithm demonstrates exceptional computational efficiency, offers high interpretability, and handles sparse feature representations typical in text data with remarkable accuracy. Its adaptability to multi-class classification scenarios made it an ideal choice for our nuanced offensive language detection task involving multiple subtasks.

By using robust feature engineering, strategic regularization, comprehensive evaluation metrics, and systematic hyperparameter tuning, we developed a Logistic Regression model capable of effectively detecting and categorizing offensive language across multiple complex classification scenarios.

Testing:

For each subtask, we employed a systematic approach for testing, predicting, and evaluating our model to ensure its effectiveness. The testing process involved loading a test dataset, where text data was cleaned through preprocessing and transformed into TF-IDF features to make it compatible with the trained model. Using this processed test data, the model generated

predictions, which were saved to a CSV file for subsequent analysis. This structured workflow ensured that the predictions were readily available for evaluation and further application in the project.

Evaluation:

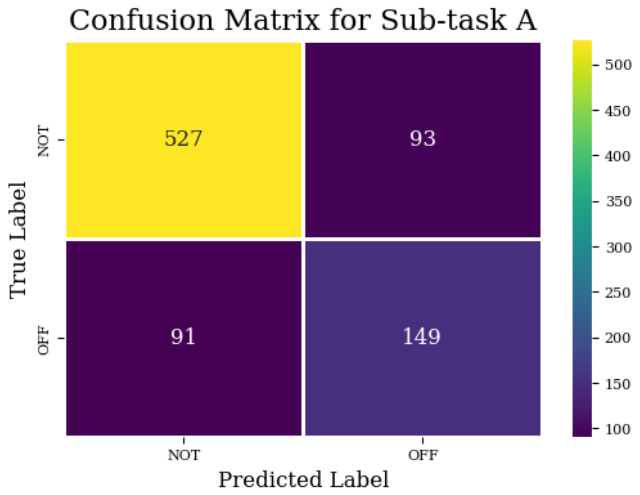
For evaluation, we employed a comprehensive framework utilizing both the labels file and the predictions file to assess the model's performance. The true labels, extracted from the labels file, were aligned with the predicted labels by matching their unique IDs to maintain consistency. The evaluation metrics included overall accuracy, recall, and F1 scores to evaluate the model's ability to differentiate between categories. To account for potential class imbalances, weighted averages and macro-F1 scores were calculated, providing additional insights into the model's balance across classes.

Furthermore, a confusion matrix was generated and visualized, offering a detailed representation of the classification results and highlighting the model's successes and areas needing improvement. This evaluation methodology provided a thorough analysis of the model's performance.

Results:

Sub-task A:

Metric	Value
Accuracy	0.7860
NOT Metrics (Precision/Recall/F1)	0.8528, 0.8500, 0.8514
OFF Metrics (Precision/Recall/F1)	0.6157, 0.6208, 0.6183
Weighted Metrics (Precision/Recall/F1)	0.7866, 0.7860, 0.7863
Macro F1 Score	0.7348



Sub-task A showed moderate performance. The model achieved an accuracy of 0.786, correctly classifying most samples. However, this does not ensure equal performance in distinguishing offensive tweets from non-offensive ones.

The confusion matrix reveals that out of 620 actual NOT tweets, the model correctly classified 527, resulting in a high recall of 0.85 for the NOT class. This means the model successfully identified most actual NOT tweets. Additionally, the precision of 0.8528 for the NOT class indicates that a high proportion of predictions labeled as NOT were correct. The F1 score of 0.8514 for the NOT class demonstrates a balanced performance in terms of precision and recall.

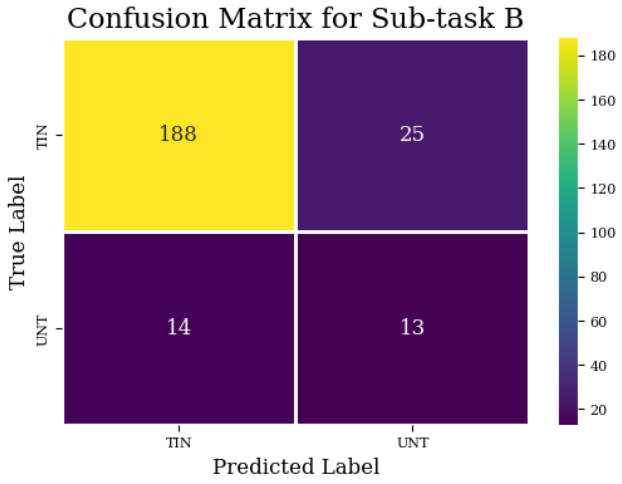
For the OFF class, the confusion matrix shows that out of 240 actual OFF tweets, the model identified only 149 correctly, leading to a recall of just over 0.62. The precision of 0.6157 for the OFF class indicates that many predictions labeled as OFF were inaccurate. The F1 score for the OFF class, at 0.6183, reflects this moderate performance in balancing precision and recall.

The overall macro F1 score for this task was 0.7348, which highlights the model's general ability to classify tweets but underscores the challenges in accurately identifying offensive content. The

confusion matrix further illustrates this imbalance, with the model performing better on the NOT class than on the OFF class.

Sub-task B:

Metric	Value
Accuracy	0.8375
TIN Metrics (Precision/Recall/F1)	0.9307, 0.8826, 0.9060
UNT Metrics (Precision/Recall/F1)	0.3421, 0.4815, 0.4000
Weighted Metrics (Precision/Recall/F1)	0.8645, 0.8375, 0.8491
Macro F1 Score	0.6530



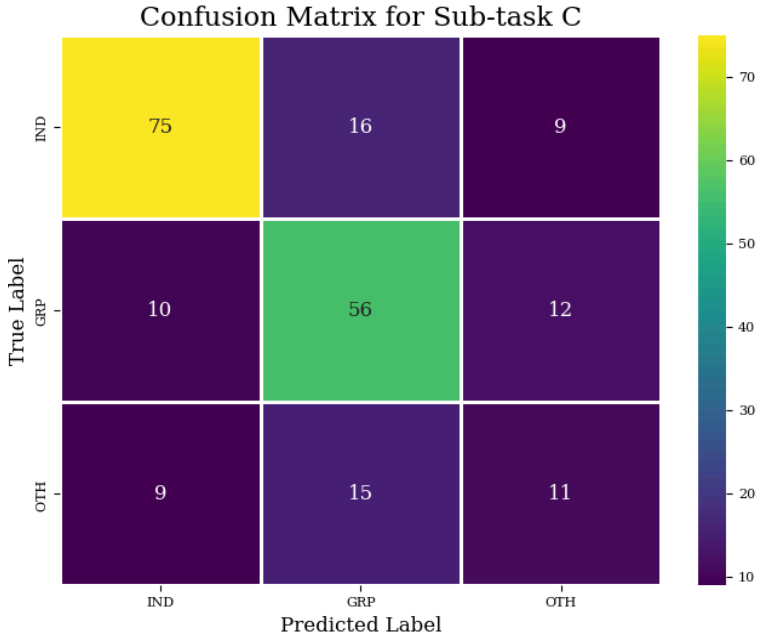
Sub-task B showed moderate performance. The model achieved an accuracy of 0.8375, correctly classifying most samples.

The confusion matrix reveals that out of 213 actual TIN tweets, the model correctly classified 188, resulting in a high recall of 0.8826 for the TIN class. This indicates the model successfully identified most actual TIN tweets. Additionally, the precision of 0.9307 for the TIN class shows that a large proportion of predictions labeled as TIN were correct. The F1 score of 0.906 for the TIN class demonstrates strong performance in balancing precision and recall.

For the UNT class, the confusion matrix highlights challenges in classification. Out of 27 actual UNT tweets, the model correctly identified only 13, leading to a recall of 48.15%. This means that less than half of the actual UNT tweets were captured. The precision of 34.21% for the UNT class further underscores the model's struggle, as many predictions labeled as UNT were incorrect. The F1 score for the UNT class, at 40%, reflects the difficulty the model had in balancing precision and recall for this minority class. The overall macro F1 score for this task was 65.30%, reflecting the model's moderate performance when averaged across both classes. The confusion matrix visually emphasizes the imbalance in performance, with the model performing significantly better on the TIN class than on the UNT class.

Sub-task C:

Metric	Value
Accuracy	0.6667
IND Metrics (Precision/Recall/F1)	0.6437, 0.7179, 0.6788
GRP Metrics (Precision/Recall/F1)	0.7979, 0.7500, 0.7732
OTH Metrics (Precision/Recall/F1)	0.3438, 0.3143, 0.3284
Weighted Metrics (Precision/Recall/F1)	0.6668, 0.6667, 0.6655
Macro F1 Score	0.5934



Sub-task C showed moderate performance. The model achieved an overall accuracy of 0.6667.

The confusion matrix provides further insight into the model's performance across the three classes. For the IND class, out of 100 actual IND tweets, the model correctly classified 75, resulting in a recall of 0.7179. The precision of 0.6437 for this class indicates that a reasonable proportion of predictions labeled as IND were correct, while the F1 score of 0.6788 reflects fairly balanced precision and recall.

For the GRP class, the confusion matrix reveals that 56 out of 78 actual GRP tweets were correctly identified, leading to a recall of 0.75. The precision of 0.7979 for this class suggests that the model performed well in predicting GRP tweets with relatively few false positives. The F1 score of 0.7732 demonstrates strong overall performance for this class.

The OTH class presented significant challenges for the model. Out of 35 actual OTH tweets, only 11 were correctly classified, resulting in a recall of 0.3143. The precision for this class, at 0.3438, highlights the model's difficulty in accurately predicting OTH tweets, as a large proportion of predictions labeled as OTH were incorrect. The F1 score of 0.3284 underscores the model's struggle to balance precision and recall for this minority class.

The overall macro F1 score for this task was 0.5934, which demonstrates the difficulty of balancing performance across all three classes. The confusion matrix emphasizes these challenges, with the model performing well on the GRP class, moderately on the IND class, and poorly on the OTH class.

Results Comparison:

Our results compared to the competition results for SemEval-2019 Task 6: Identifying and Categorizing Offensive Language in Social Media (OffensEval) are as follows:

Our Logistic Regression model for Subtask A achieved a macro F1 score of 0.7348, placing it within the 46-57 rank range (F1 range: 0.730-0.739) This performance is particularly impressive given that Logistic Regression is a much simpler model compared to deep learning approaches like CNN and BiLSTM, which achieved F1 scores of 0.800 and 0.750, respectively. While our model does not outperform these advanced architectures, it demonstrates competitive performance, which proves the effectiveness of our preprocessing steps and TF-IDF feature engineering. Furthermore, our model outperforms other machine learning baselines such as SVM (0.690)

Our Logistic Regression model achieved a macro F1 score of 0.6530 for Sub-task B, placing it within the 25-29 rank range (F1 range: 0.640-0.655) This performance highlights the strength of our simpler model, given that it performs on par with the BiLSTM model, which scored 0.660, and outperforms the SVM model, which scored 0.640. While our model does not reach the levels of the CNN model (0.690), it demonstrates competitive performance.

Our Logistic Regression model achieved a macro F1 score of 0.5934 for Sub-task C, placing it in 7th place. This performance is remarkable, as it surpasses complex models like CNN and BiLSTM, both of which scored 0.470, as well as the SVM model, which achieved 0.450.

Conclusion:

The identification of offensive language on social media is a critical task in maintaining a respectful online environment while upholding free expression. Our study leveraged the OLID dataset from SemEval-2019 Task 6, employing preprocessing techniques such as tokenization and

TF-IDF vectorization, and training Logistic Regression models for multi-level classification of offensive language.

Our findings demonstrate that the chosen methods yielded varying performance across Sub-tasks A, B, and C, with clear strengths and areas for improvement. Sub-task A achieved a weighted precision of 0.7866 and a weighted recall of 0.7860, indicating balanced performance in classifying offensive (OFF) and non-offensive (NOT) tweets, with a macro F1 score of 0.7348 reflecting stronger results for the majority class (NOT). Sub-task B demonstrated stronger overall performance, with a weighted precision of 0.8645 and a weighted recall of 0.8375, but its macro F1 score of 0.6530 revealed an imbalance, as the model excelled at identifying targeted (TIN) tweets but struggled with untargeted (UNT) tweets. Sub-task C presented the greatest challenge, with a weighted precision and recall of 0.6668 and 0.6667, respectively, and a macro F1 score of 0.5934 highlighting significant discrepancies across classes, particularly with the model's difficulty in handling the minority class (OTH).

These results underscore the complexity of offensive language detection, especially in nuanced tasks such as identifying specific targets. Future work could explore the integration of more advanced machine learning models, such as ensemble methods or deep learning approaches, to enhance performance. Additionally, expanding the dataset with more diverse examples and employing contextual embeddings could improve the identification of offensive language targets.

In conclusion, this study highlights the potential and challenges of offensive language identification, providing a foundation for future research aimed at creating safer and more inclusive digital communication spaces.

Acknowledgements:

We would like to express our heartfelt gratitude to Dr. Marcos Zampieri for his invaluable support, encouragement, and guidance throughout the semester. His expertise and insightful feedback have been instrumental in shaping the direction of this project. We also extend our sincere thanks to our Teaching Assistant, Sadiya Sayara Chowdhury Puspo, for her support and constructive feedback, which have greatly contributed to the successful completion of this work.

References:

Zampieri, M., Malmasi, S., Nakov, P., Rosenthal, S., Farra, N., & Kumar, R. (2019). Predicting the type and target of offensive posts in social media. *In Proceedings of the NAACL. Association for Computational Linguistics.*

Zampieri, M., Malmasi, S., Nakov, P., Rosenthal, S., Farra, N., & Kumar, R. (2019). SemEval-2019 Task 6: Identifying and Categorizing Offensive Language in Social Media (OffensEval). *In Proceedings of the 13th International Workshop on Semantic Evaluation* (pp. 75–86).

Michael Wiegand, Melanie Siegel, and Josef Ruppenhofer. 2018. Overview of the GermEval 2018 shared task on the identification of offensive language. *In Proceedings of the GermEval 2018 Workshop (GermEval).*