International Journal of Data Science and Analytics
https://doi.org/10.1007/s41060-024-00607-9

REGULAR PAPER

Towards a cyberbullying detection approach: fine-tuned contrastive self-supervised learning for data augmentation

Lulwah M. Al-Harigy[1,2] · Hana A. Al-Nuaim[1] · Naghmeh Moradpoor[2] · Zhiyuan Tan[2]

Received: 26 December 2023 / Accepted: 4 July 2024

Abstract

Cyberbullying on social media platforms is pervasive and challenging to detect due to linguistic subtleties and the need for extensive data annotation. We introduce a Deep Contrastive Self-Supervised Learning (DCSSL) model that integrates a Natural Language Inference (NLI) dataset, a fine-tuned sentence encoder, and data augmentation to enhance the understanding of cyberbullying's nuanced semantics and offensiveness. The DCSSL model effectively captures contextual dependencies and the varied semantic implications inherent in cyberbullying instances, addressing the limitations of manual data annotation processes when compared against established models such as BERT and Bi-LSTM. Our proposed model registers a significant improvement, achieving a macro average F1 score of 0.9231 on cyberbullying datasets, highlighting its applicability in environments where manual annotation is impractical or unavailable.

Keywords Cyberbullying Detection · Deep Contrastive Self-Supervised Learning · Data Augmentation · Natural Language Inference · Offensive Content Detection

1 Introduction

In the last decade, the exponential increase in internet users [1], aided by wider service coverage, lower costs, and diverse electronic devices and applications, has significantly enhanced access to the online world. The rise of social media has simplified communication but also enabled harmful social behaviors such as the dissemination of insulting comments and news, often anonymously [2]. This anonymity has exacerbated abuse, especially against vulnerable groups.

Bullying has long existed, traditionally involving direct, face-to-face insults. However, the rise of social media and increased internet accessibility have transformed bullying, enabling cyberbullying to occur at any time and from anywhere, expanding its reach and impact.

Because of the widespread reach of online platforms, cyberbullies can bring about intense harm to their victims, with a potential audience of hundreds to sometimes hundreds of thousands of people, eclipsing the traditional schoolyard bullying, which may have only a few witnesses. Cyberbullying differs in its potential for anonymity, a large audience, and greater physical distance from the victim, facilitating malicious behavior without consequences for the perpetrator.

The COVID-19 pandemic saw a sharp rise in cyberbullying as lockdowns led to increased online activity for work and education. With cyberbullying becoming widespread and causing significant societal harm, it's critical to utilize automatic detection methods to curb such behavior online [2]. Detecting online bullying is complicated by factors such as social media specificities, topic dependencies, and the diversity of manually crafted features.

Analyzing a given text requires capturing the underlying meaning behind the text, such as the existing semantic, syntactic, and spatial relationships. Learning representative features automatically using deep learning models efficiently captures the contextual semantics and word order arrangement to build solid and excessive predictive models [3]. The nature of texts containing bullying or insults differs from other texts in its dependence on context; therefore, to annotate datasets containing bullying or offensive sentences, it is important to take into account the whole context of the sentence instead of the individual words themselves.

✉ Lulwah M. Al-Harigy
lalharigy@kau.edu.sa

[1] Faculty of Computing and Information Technology, King Abdulaziz University, Jeddah, Saudi Arabia

[2] School of Computing, Engineering and the Built Environment, Edinburgh Napier University, Edinburgh, UK

Published online: 17 July 2024

Datasets can be annotated manually or automatically. Manual labeling, done by experts or through crowdsourcing, ensures high-quality data but is costly and time-consuming, especially for large datasets [4]. This challenge has led to the development of alternatives. Self-supervised methods use the data itself for supervision, avoiding expensive manual annotation. Automated labeling can involve self-annotation or self-supervised learning (SSL). Self-annotation uses hashtags or offensive words to label data; however, this technique may result in inaccurate characterization of the data since the context is not considered [4].

Deep neural networks have shown their comprehensive successes in learning from large-scale labeled datasets; however, this success hinges on the availability of a large number of labeled examples that are expensive to collect [5].

SSL has shown remarkable results in representation learning [6], and research has focused on SSL as a way to label huge datasets in a cost-effective and timely manner [5]. The goal of SSL is to learn to extract representations of input using a large amount of unlabeled data to solve downstream tasks which often have only a few labeled examples [7]. Self-supervised learning methods have been widely studied to close the gap with supervised learning and eliminate the time and cost of labeling a large amount of data [5].

Contrastive learning is an approach used to differentiate between data by using similarity measurements to pull that which is similar close to each other and push that which is dissimilar away. Contrastive self-supervised learning (CSSL)—a subcategory of SSL between self-supervised and supervised learning—fills the gap between them [6]. CSSL aims to generate a new representation of the original data to augment data by calculating the similarity between examples to determine whether two examples are similar (creating positive pairs) or dissimilar (creating negative pairs) [8–10].

This work is an extension of our previous work [11] for detecting cyberbullying using augmented datasets. In both works (the previous and current work), we utilized CSSL approach to increase the training set using data augmentation. The CSSL-based model is used to generate new representations and find augmented data for the small labeled dataset (OLID) from the large-scale dataset (SOLID) by calculating the similarity between labeled and unlabeled examples; thus, the augmented data are unlabeled. Each example in the augmented data will be labeled taking into account the label of its similar example from the small labeled dataset and a decision from additional algorithm. In both works, we integrated the supervised SimCSE ($\mathrm{SimCSE}_{\mathrm{supervised}}$) for data augmentation and our proposed model parallel BERT + Bi-LSTM for detecting cyberbullying using the augmented data. The key differences between the two works are that in the previous work [11], we made the decision to choose and annotate the augmented data using CSSL-based model represented by $\mathrm{SimCSE}_{\mathrm{supervised}}$ model and the parallel BERT + Bi-LSTM model. In this work, we create a new NLI dataset using labeled dataset (OLID) and use it to optimize $\mathrm{SimCSE}_{\mathrm{supervised}}$ to force it to maintain both the semantic and offensive meanings when generating new representations and calculating similarity scores.

This research builds upon previous work in cyberbullying detection by introducing an enhanced approach that leverages augmented data through CSSL. This significantly expands the available dataset, enhancing the model's ability to learn and generalize from a larger corpus of examples.

1.1 Research questions.

To achieve the objectives of this research, we aim to assess the impact of utilizing pre-trained Deep Learning algorithms like BERT and augmented datasets using CSSL on the performance of cyberbullying detection via a social media platform such as Twitter. Thus, this paper will address the following research questions:

- RQ1: How does the performance of the proposed fine-tuned Deep Contrastive Self-Supervised Learning DCSSL model based on pre-trained models compare with that of the baseline models for cyberbullying detection on social media?
- RQ2: How does the performance of the proposed model using augmented data compare with a manually labeled dataset for cyberbullying detection?
- RQ3: How does the performance of CSSL for annotating dataset compare with the manually labelled dataset?

1.1 Contributions

The contributions of the current paper are:

1- Evaluation of the effectiveness of the fine-tuned Deep Contrastive Self-Supervised Learning (DCSSL) model for cyberbullying detection on social media. The comparison of the proposed model's performance against baseline models will aid in determining its significance as an improvement over existing methods. This contribution is aligned with RQ1.

2- Assessment of the impact of utilizing augmented data to train the proposed model. The comparison between the performance of the model trained on augmented data and that trained on a manually labeled dataset will help ascertain whether augmented data contributes to enhancing the performance of cyberbullying detection models. This contribution is aligned with RQ2.

3- Evaluation of the effectiveness and accuracy of a CSSL-based model in annotating a manually labelled dataset.

The comparison of the annotations made by the CSSL-based model with the ground truth labels provided by human annotators is essential for determining the accuracy and reliability of the CSSL-based model's annotations. The findings can offer valuable insights for practical applications, especially in scenarios where large-scale data annotation is required, as manually labelled data for cyberbullying is expensive and time-consuming to create. This contribution is aligned with RQ3.

This paper is organized as follows: Sect. 2 will describe the related works in the CSSL and cyberbullying detection fields. Section 3 will explain the different components of the proposed system. The results analysis will be discussed in Sect. 4, while Sect. 5 will include the conclusion and future work.

2 Literature review

Given the focus of the research in this paper, this section reviews the existing work related to 1) CSSL-based methods for data annotation using data augmentation, especially text augmentation methods, and 2) cyberbullying detection techniques with a particular focus on transformers such as BERT-based approaches as we utilized BERT for detecting cyberbullying.

2.1 CSSL for data augmentation

Self-supervised learning (SSL) enables a network to learn meaningful data representations through pretext tasks based on the input data itself, such as image inpainting, predicting rotations, synonym replacement, and sentence similarity. These tasks eliminate the need for extensive human-annotated data. Contrastive self-supervised learning (CSSL) generates these pretext tasks by training the model to compare the original sample (anchor) with similar (positive) and dissimilar (negative) samples from unlabeled data [7]. Data augmentation is crucial in CSSL, helping to minimize the distance between similar samples and maximize the distance between dissimilar ones [6].

In this subsection, we review existing work on automated data labeling, particularly using CSSL for cyberbullying detection to highlights the key differences between these models and our proposed approach. CSSL trains machines to recognize similarities and differences among samples, helping them learn patterns and make predictions. While most research has applied CSSL to computer vision, focusing on image augmentations [5, 6, 9, 15], we explore its use with text augmentation. However, we also consider some studies that applied CSSL to images.

Image augmentation plays a crucial role in contrastive learning for computer vision tasks, significantly differing from text augmentation techniques. Unlike text, where augmentations might alter meaning, image augmentations like cropping, rotating, or color jittering can create positive pairs while preserving the image's core content. Several studies explored diverse image augmentation strategies for contrastive learning: Wang and Qi [5] proposed Contrastive Learning with Stronger Augmentations CLSA, a framework that utilizes a random combination of 14 augmentation types. Miyai et al. [6] introduced Positive or Negative Data Augmentation PNDA, an augmentation strategy that considers image semantics for a more targeted approach. Xiao et al. [9] presented Leave-one-out Contrastive Learning LooC, a framework designed for multi-augmentation contrastive learning. In contrast, Chen et al. [15] proposed simple framework for contrastive learning of visual representations SimCLR, a framework that simplifies contrastive learning by removing the need for specialized architectures or memory banks.

Augmenting text requires more attention to context compared to image augmentation. Inappropriate text augmentation, particularly for offensive language, can lead to misleading meanings and hinder detection performance. Unlike image cropping, text augmentation methods often focus on preserving context through techniques like word substitution, synonym replacement, back-translation, or calculating similarity scores. However, caution is advised when using contrastive self-supervised learning (CSSL) with text augmentation for offensive language tasks due to potential meaning shifts.

Several studies have explored text augmentation for contrastive learning. Different models like SimCSE [12], CERT [10], CLEAR [16], DeCLUTR [17], DualCL [18], SDA [19], and ContrastNet [20] have been proposed with varying degrees of success and efficiency in applying contrastive learning to NLP tasks. These approaches aim to enhance sentence representation quality by maximizing agreement between augmented versions of the same sentence while keeping them distinct from other sentences in the batch.

Contrastive learning has emerged as a powerful technique for learning sentence representations for various NLP tasks.

Fang et al. [10] propose CERT, a contrastive self-supervised learning (CSSL) approach that utilizes back-translation to create augmented sentences for pre-training a language encoder (e.g., BERT). This approach improves sentence-level understanding by distinguishing between augmented sentences derived from the same source. They used back-translation on a single dataset for augmentation, while our approach calculates similarity scores to match labeled examples with unlabeled ones, creating augmented data from two datasets. While CERT focuses on generating similar representations, Wu et al. [16] present CLEAR, a method that combines contrastive learning with masked language modeling (MLM) for pre-training. CLEAR aims for robust sentence representations less susceptible to noise by employing various augmentation techniques like word deletion or reordering. However, these modifications can alter the meaning of sentences, particularly those expressing sentiment. They used sentence-level modification techniques (deletion, reordering, substitution) on a single dataset for augmentation, while we created augmented data by finding similar examples between labeled and unlabeled data using similarity scores. Giorgi et al. [17] address the challenge of unsupervised learning with DeCLUTR, a framework that leverages contrastive learning to generate high-quality sentence embeddings without labeled data. DeCLUTR trains a sentence encoder to minimize the distance between embeddings of similar textual segments from the same document. They combined CL with MLM for learning sentence embeddings using a single dataset for augmentation. In contrast, we created augmented data through similarity scores between labeled and unlabeled data.

Q. Chen et al. [18] propose Dual Contrastive Learning (DualCL), a supervised contrastive learning approach for text classification. DualCL aims to learn informative representations for both text data and labels. The approach utilizes label-aware data augmentation, where the model identifies informative variations of a sentence based on its relationship to its label and other similar labeled examples within the training data. On the other hand, contrastive learning can also be applied in unsupervised settings for sentence representation learning. They calculated similarity scores within a single dataset by comparing a sentence to its positive pair, its label, and the label's positive pair. In contrast, we generate augmented data by calculating similarity scores between labeled and unlabeled examples. Mao et al. [19] proposed Simple Discrete Augmentation (SDA), a model that employs three discrete augmentation methods (punctuation insertion, affirmative auxiliary, and double negation) to create diverse representations for each sentence. This approach is suitable for scenarios where labeled data is scarce. They used three methods within a single dataset to create sentence variations for augmentation, while we generated augmented data by calculating similarity scores between labeled and unlabeled examples.

Furthermore, contrastive learning demonstrates promise in few-shot text classification tasks, where only a handful of labeled examples per class are available. J. Chen et al. [20] proposed ContrastNet, a framework that utilizes contrastive learning to obtain discriminative text representations. This is crucial for few-shot classification (very few training examples per category), as similar representations for similar classes can lead to misclassification during prediction. They classified text with few-shot learning while we addressed limited data by creating similar examples from unlabeled data using cosine similarity.

Various approaches to Contrastive Semi-Supervised Learning (CSSL) have been proposed in the literature, including similarity scores [18], back translation [10], and discrete augmentation (PI, AA, and DB) [19]. Our work differs from previous approaches by using a sentence encoder (SimCSE) trained with contrastive self-supervised learning (CSSL) to create new data representations. This approach preserves both semantic and offensive meanings in the augmented data. We further refine the encoder by retraining it on a newly created Natural Language Inference (NLI) dataset. Unlike prior works, we use SimCSE to generate new representations for both labeled and unlabeled data. Positive pairs are created by matching examples from the labeled dataset with those in the unlabeled set, allowing us to assign labels to these newly generated examples.

2.2 Pre-trained model for cyberbullying detection

Cyberbullying is a serious issue on social media. Natural language processing (NLP), especially deep learning models like BERT, has shown promise in identifying cyberbullying. BERT, in particular, excels at understanding context and nuances of language, making it effective in recognizing abusive language and hate speech. By leveraging NLP and machine learning, BERT can identify patterns associated with cyberbullying and achieve high accuracy in detecting such content.

Most cyberbullying detection work focuses on understanding sentence context to create a whole-sentence representation. Each word's meaning can depend on its position and the sentence's underlying intent. Previous research showed deep learning's effectiveness for cyberbullying detection but typically used a single model for data representation. Our research improves this by using multiple models to create richer sentence representations, enhancing word understanding based on their context, and deepening semantic analysis. Transformer encoders or decoders are pre-trained on large text corpora by solving self-supervised tasks like predicting masked tokens [21], generating the next sentence [22], or denoising corrupted tokens [10, 24]. BERT, trained to predict missing words based on surrounding context, has become a popular choice for low-resource text classification tasks [7]. Many researchers have leveraged pre-trained models like BERT [25–27] and ALBERT [28] for detecting offensive language in social media content.

Several studies explore various deep learning approaches for cyberbullying detection on social media platforms. Paul and Saha [25] introduced CyberBERT, a fine-tuned BERT model that addresses the challenge of imbalanced data in cyberbullying detection. They used knowledge distillation to shrink a large model (BERT) into a smaller one while we used the full power of BERT-based version combined with a deep learning classifier Bi-LSTM to generate different representations for the sentence. Elsafoury et al. [26] reinforced the effectiveness of contextual language models like BERT in their survey, highlighting the need for further research on BERT pre-training tailored for cyberbullying tasks. While BERT dominates current research, Kumar and Sachdeva [2, 3] proposed alternative architectures like capsule networks (CapsNets) and Bi-directional GRUs (Bi-GRUs) with attention for text representation and classification, demonstrating promising results. They tackled cyberbullying across various modalities (text, visuals, infographics) using CapsNet and ConvNet architectures while we focused on text data.

Furthermore, researchers explore techniques to address limitations in real-world data. Nouri [29] proposes a dual training approach for Offensive Span Detection (OSD) that utilizes GPT-2 for generating synthetic offensive examples to augment training data for fine-tuning a BERT model. This approach tackles the issue of limited labeled data for offensive language detection. They used GPT-2 to generate synthetic text for data augmentation, while we focused on finding similar examples from unlabeled data using labeled data and cosine similarity.

Cyberbullying detection can extend beyond text analysis. Gonzalez-Pizarro and Zannettou [30] presented a model using Contrastive Language-Image Pre-training (CLIP) to detect hate speech in text and images, highlighting the prevalence of hateful imagery and the importance of multimodal methods. They addressed hate speech in text and images using cosine similarity, while we focus on cyberbullying detection in text by combining BERT and Bi-LSTM with augmented data.

Bhatia et al. [31] introduced a custom Bi-LSTM model with GloVe embeddings that incorporates a slang corpus to capture informal language specific to social media. They compared their approach to BERT, demonstrating its potential for improved precision and accuracy in cyberbullying detection. Their model used Bi-LSTM with GloVe for text embedding since Bi-LSTM lacks a built-in embedding layer, unlike BERT. Our approach leverages BERT combined with Bi-LSTM to create richer sentence representations.

Guo et al. [27] proposed Augmented BERT, a method that combined data augmentation techniques with BERT for cyberbullying detection. Their approach is particularly useful for scenarios with limited labeled data, as they employ generative adversarial networks (GANs) and autoencoders to increase the training data size. They used GAN-based techniques for data augmentation and relied solely on BERT for cyberbullying detection with a single sentence representation. In contrast, our work combines BERT with Bi-LSTM to create richer sentence representations.

3 The proposed model

For this research, we propose a model to detect cyberbullying by leveraging augmented data using a CSSL-based model represented by the fine-tuned SimCSE$_{\text{supervised}}$. This model learns new sentence representations and computes cosine similarities between sentences to identify positive pairs (entailment) and hard negative pairs (contradiction) for each sentence.

3.1 Research methodology

This section elaborates on the various components of the proposed model for detecting cyberbullying utilising augmented datasets and covers the methodological aspects of the employed methods. The approaches employed in the proposed model are explained, encompassing dataset selection, preprocessing techniques, the models utilised for training, and the augmentation technique employed to expand the training set.

To be able to answer the research questions, the following methodology was carried out for this research (see framework in Fig. 1).

Figure 1 illustrates the workflow steps for the proposed model, employing SimCSE$_{\text{optimized}}$ for data augmentation. While the figure provides a high-level overview, it may not capture all the intricate details involved in each step. Consequently, the specific steps for executing the methodology are elaborated upon as follows:

1. Identify the available datasets in the literature with particular attention to those including emojis (discussed in Sect. 3.1.1);
2. Choose a small labelled dataset (OLID) to train the proposed model (discussed in SubSect. 3.1.1.1);
3. Choose a large-scale unlabelled dataset (SOLID) for cyberbullying to create augmented data (discussed in SubSect. 3.1.1.2);
4. Clean and pre-process the chosen datasets (discussed in Sect. 3.1.2); tokenize, replace mentions @, hyperlinks URL, hashtags # and retweet RT;
5. Integrate the parallel BERT + Bi-LSTM classification model for cyberbullying detection; generate one representation for each sentence by combining two different outputs from BERT and Bi-LSTM;
6. Create a new Natural Language Inference (NLI) dataset by encoding the small cyberbullying dataset (OLID) using SimCSE$_{\text{supervised}}$ to find the positive and hard_negative pairs for each example in OLID (discussed in Sect. 3.4);
7. Create SimCSE$_{\text{optimized}}$ by retraining SimCSE$_{\text{supervised}}$ on the dataset from Step (6) (discussed in Sect. 3.5);

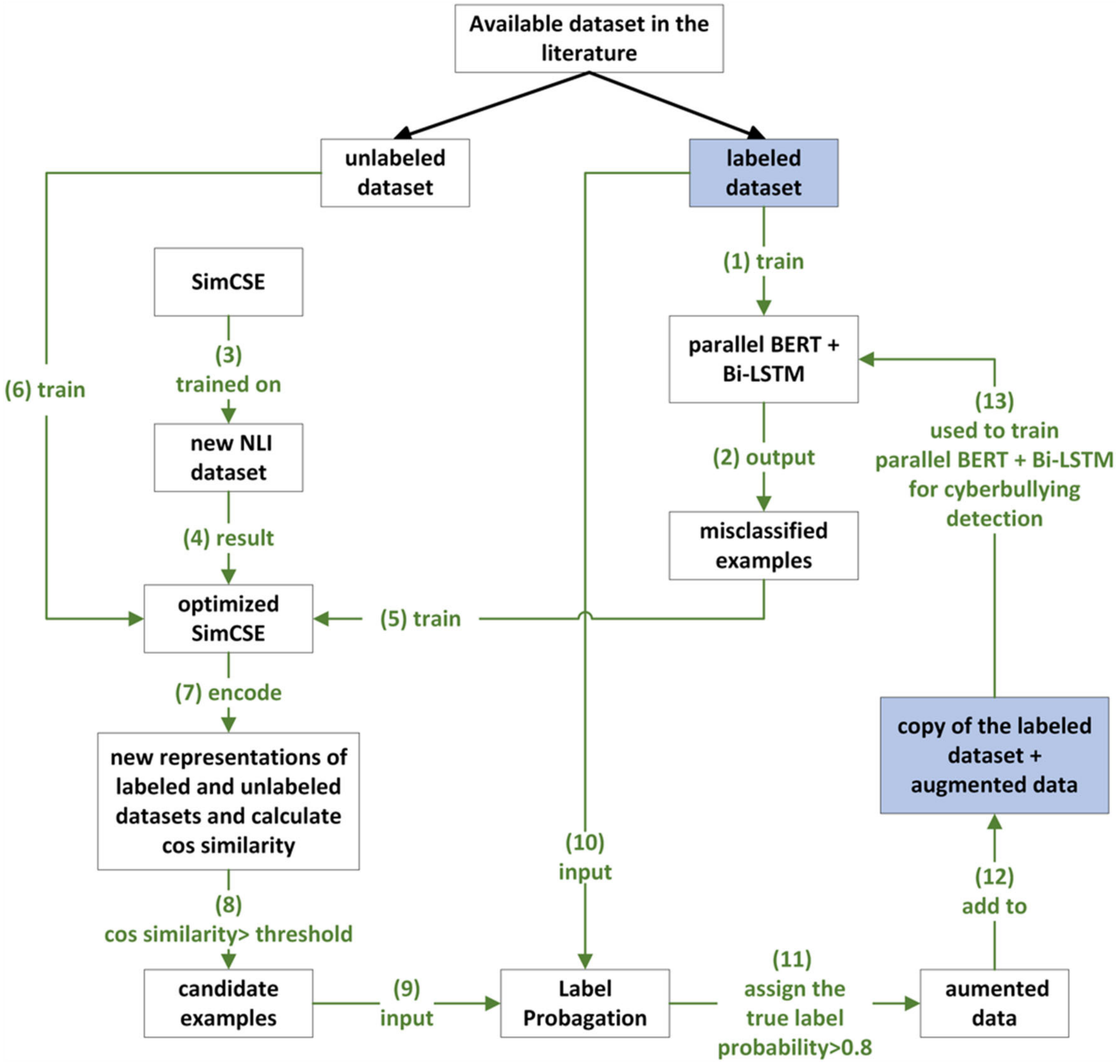

Available dataset in the literature
unlabeled dataset
labeled dataset
(1) train
SimCSE
(3) trained on
parallel BERT + Bi-LSTM
(6) train
(13) used to train parallel BERT + Bi-LSTM for cyberbullying detection
new NLI dataset
(2) output
misclassified examples
(4) result
optimized SimCSE
(5) train
(7) encode
new representations of labeled and unlabeled datasets and calculate cos similarity
copy of the labeled dataset + augmented data
(10) input
(12) add to
(8) cos similarity> threshold
candidate examples
(9) input
Label Probagation
(11) assign the true label probability>0.8
aumented data

Fig. 1 The methodology framework

8. Train the model parallel BERT + Bi-LSTM from [11] using the dataset from Step (2) (discussed in Sect. 3.5.1);
9. Find the misclassified examples OLID_m resulting from training the model on Step (8) (discussed in Sect. 3.5.2); misclassified examples are the offensive (OFF) or not offensive (NOT) examples classified incorrectly by the model parallel BERT + Bi-LSTM;
10. Use $\text{SimCSE}_\text{optimized}$ for encoding the misclassified examples OLID_m from Step (9) and the SOLID training set from Step (5) (discussed in Sect. 3.5.3);
11. Generate new representations for the OLID_m dataset from Step (9) and the SOLID from Step (3) using $\text{SimCSE}_\text{optimized}$ (discussed in Sect. 3.5.3);
12. Calculate the cosine similarities between the new representations result from Step (11) using the $\text{SimCSE}_\text{optimized}$ model (discussed in Sect. 3.5.4);
13. If the cosine similarity for the examples from Step (12) is greater than *threshold*, add the example to the augmented examples;
14. Assign the pseudo labels for the augmented examples using the Label Propagation (LP) algorithm using predefined *threshold* (discussed in Sect. 3.5.5);
15. Add the augmented examples to the OLID training set;
16. Train the model parallel BERT + Bi-LSTM from [11] to detect cyberbullying using the augmented cyberbullying dataset from Step (15) (discussed in Sect. 3.5.6);
17. Conduct different experiments to evaluate the performance of the proposed model (discussed in Sect. 4).

3.1.1 Dataset description

We leverage a dataset from platform X, known for high cyberbullying incident reports as one of the top five [21, 32] and its frequent use in related research for baseline comparisons. This choice allows for comparison with existing work. To preserve tweet integrity, we use unprocessed data including emojis. Training utilizes two datasets: labeled OLID [13] and unlabeled SOLID [14], the only large-scale options for cyberbullying ad offensive detection with emojis and established benchmarks for offensive language identification (SemEval-2019 [33], SemEval-2020 [34]). We focus solely on level A of these datasets, distinguishing offensive ("OFF") from non-offensive ("NOT") sentences.

3.1.1.1 OLID dataset The OLID dataset contains a total of 14,100 entries, of which 13,240 are allocated for training purposes and the remaining 860 set aside for testing. The OLID dataset was used in this research as a seed to train the proposed model and find the misclassified examples.

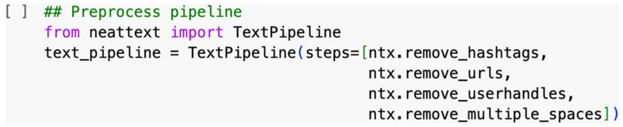
3.1.1.2 SOLID dataset In contrast to OLID, the SOLID dataset comprises significantly more data, with 9,089,140 examples for training and 5,993 designated for testing. The training portion of the SOLID dataset was prepared using a democratic co-training approach, whereas the test portion received expert manual labeling. Approximately one million tweets, originally compiled by Al-Harigy et al. [11], were utilized for the SOLID dataset since the competition organizers only provided tweet IDs and not the dataset itself.

In our research, we utilized the SOLID dataset to identify sentences that could augment the OLID dataset. We assigned pseudo labels to this augmented data based on similarity scoring and assessed the outcomes using the test set from SOLID. Subsequently, we incorporated the augmented data into the OLID dataset to enhance its size and variability.

3.1.2 Data pre-processing and cleaning

Pre-processing refines the dataset for model training. This includes noise reduction and size optimization by removing irrelevant content (stop words, punctuation, URLs, etc.) to focus models on important words. Common pre-processing steps applied in our model include stop word removal, tokenization, and lowercasing (references omitted). The following are the most widely used pre-processing steps [4] to clean the dataset which were used in our proposed model:

1- Remove mentions @, hyperlinks URL, hashtags # and retweet RT: this is the most commonly used pre-processing technique to reduce noise. This pre-processing step is used in this research to clean the chosen dataset using the Python code presented in Fig. 2. The figure presents the four steps used to clean the datasets by removing hashtags, urls, user handles (@), and multiple spaces between words.



```
[ ] ## Preprocess pipeline
    from neattext import TextPipeline
    text_pipeline = TextPipeline(steps=[ntx.remove_hashtags,
                                        ntx.remove_urls,
                                        ntx.remove_userhandles,
                                        ntx.remove_multiple_spaces])
```

Fig. 2 Code for the pre-processing steps used in the proposed model

2- Tokenization: Our proposed model integrates two different architectures for cyberbullying detection—BERT and Bi-LSTM. BERT comes with its own tokenizer, while Bi-LSTM requires an embedding layer. For Bi-LSTM, we utilized GloVe to tokenize sentences before feeding them into the model.

The cleaned datasets are then used to train the parallel BERT + Bi-LSTM and SimCSE models.

3.2 The parallel BERT + Bi-LSTM model for detecting cyberbullying

The proposed model consists of three components which work together to detect cyberbullying. One of the components is the parallel BERT + Bi-LSTM model, proposed in our previous work [11], that combines BERT and Bi-LSTM for cyberbullying detection. Sentences are fed into parallel BERT and Bi-LSTM paths. BERT's CLS token and Bi-LSTM's output are then processed by separate FCNNs. It generates two different representations for each example in the OLID training set to ensure understanding of the underlying meaning of the sentence. The model leverages both representations by summing their logits element-wise before feeding them to a SoftMax layer for final prediction. Figure 3 illustrates the architecture.

By ensembling the logits from two architectures, we can leverage the complementary strengths of both models and improve the learning. The combined predictions provide a more robust and comprehensive understanding of the input text, resulting in improved learning capability. This approach allows us to capture a wider range of linguistic patterns and enhance the overall performance of the text classification model.

3.3 Simple contrastive learning of sentence embeddings (SimCSE)

To annotate unlabeled data using CSSL for data augmentation, we employed SimCSE, a model known for its effective sentence embeddings. SimCSE was selected due to its superior performance over other CSSL techniques, including cropping, word deletion, synonym replacement, and MLM

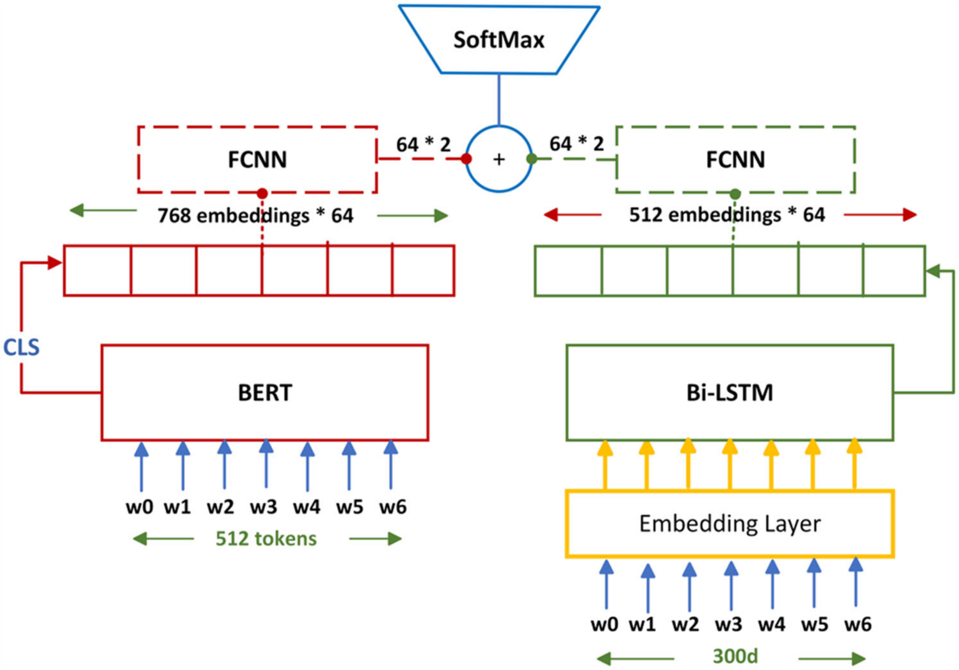


Fig. 3 Parallel BERT + Bi-LSTM model [11]

with BERT$_{\text{base}}$. These techniques were evaluated based on how well their sentence similarity scores correlated with human judgment. Unlike most studies that generate positive pairs using a single dataset, our approach uses two different datasets. This allows us to leverage labeled data to assign labels to unlabeled examples by calculating similarity scores between them, integrating a CSSL model tailored for this purpose.

In our proposed model, we incorporated SimCSE$_{\text{supervised}}$ using the training objective represented in Eq. (1) used in this work for a mini-batch of N sentences where T is a temperature hyperparameter, h_i is the representation of the example x_i, (h_i^+, h_i^-) are the representations of entailment x_i^+ and contradiction x_i^- examples, respectively, and $sim\left(h_i,\, h_i^+\right)$ is the cosine similarity, is as follows [12]:

$$\ell_i = -\log \frac{e^{sim\left(h_i, h_i^+\right)/T}}{\sum_{j=1}^{N}\left(e^{sim\left(h_i, h_j^+\right)/T} + e^{sim\left(h_i, h_j^-\right)/T}\right)} \tag{1}$$

SimCSE$_{\text{supervised}}$ approach didn't preserve offensiveness, so we optimized it. To enhance the model and avoid overfitting and underfitting, augmented examples must exhibit semantic similarity and accurate downstream labeling. Semantic similarity is ensured, but accurate labels for unseen examples are challenging. To address this, we focused on denoising augmentation labels by analyzing similarity scores within the labeled OLID dataset [13].

Table 1 shows examples of cosine similarity results between sentences from the OLID training set [13], which labelled as OFF or NOT, using SimCSE$_{\text{supervised}}$ [12]. SimCSE$_{\text{supervised}}$ calculates the similarity score between pairs of sentences. For instance, **Sent1**, labelled as OFF in OLID, and its most similar sentences (**Sent11**, **Sent12**, and **Sent13**) have scores above 0.9, indicating semantic similarity. However, they differ in offensiveness as indicated by their labels (**Sent11**, **Sent12**, and **Sent13** are labelled as NOT in OLID). The same in **Sent2** and **Sent21**, they have different labels as **Sent2** is labeled as NOT while **Sent21** is labeled as OFF. Sometimes, SimCSE$_{\text{supervised}}$ bases similarity on shared topics, such as 'borrow' in **Sent1**, **Sent11**, **Sent12**, and **Sent13**, or 'NFL' in **Sent2** and **Sent21** without considering offensiveness.

3.4 Creating a new NLI dataset

In order to force SimCSE to consider the offensive meaning, we fine-tuned SimCSE to adapt to the embedding space of the downstream task by using cosine similarity and the label between each two-sentence pair. To do that, we propose a method to reconstruct a new NLI dataset from the OLID dataset using the same structure of the NLI dataset, such that:

- Positive pairs (entailment) are the ones that have a high similarity score greater than the threshold and have the same label from the OLID training data;

Table 1 Calculating cosine similarity between OLID sentences using SimCSE

Sent	Most similar sentences	Similarity Score	Similar in semantics	Similar in offensiveness
Sent1: 'Let me borrow yo shit first ' (**OFF**)	**Sent11:** 'Lemme borrow that' (**NOT**)	0.92	Y	N
	Sent12: 'You can borrow mine' (**NOT**)	0.94		
	Sent13: 'Oooh, can i borrow when you're done? ' (**NOT**)	0.98		
Sent2: 'I hope the NFL folds!' (**NOT**)	**Sent21:** 'To hell with the NFL' (**OFF**)	0.95	Y	N

- Negative pairs (contradiction) are those that either have a low similarity score or have contradicting labels from the OLID training data.

Figure 4 demonstrates the algorithms used for the creation of the new NLI dataset leveraging SimCSE$_{\text{supervised}}$ for calculating the cosine similarity between the examples in the OLID training set and the correct label of the examples.

Figure 5 demonstrates the process of retraining SimCSE$_{\text{supervised}}$ on the new NLI dataset to create SimCSE$_{\text{optimized}}$ and the loss function which will be used by SimCSE$_{\text{optimized}}$ to generate the new embeddings for the sentences but with different definitions for the positive and negative pairs. The symbols $\boldsymbol{h}$, $\boldsymbol{h}^{+}$, and $\boldsymbol{h}^{-}$ demonstrate the representations of the original sentence, the positive pair, and the hard negative, respectively.

Figure 6 shows a screenshot of the new NLI dataset created using SimCSE$_{\text{supervised}}$ on the OLID training set. SimCSE calculates the cosine similarity for each sentence (Column **sent0**) to find its positive pair (Column **sent1**) and negative pair (Column **hard_neg**). As shown in Fig. 7, the cosine similarity scores between **sent0** and **sent1**, and **sent0** and **hard_neg**, are nearly the same—0.71 and 0.70, respectively. However, in the new NLI dataset, **sent1** is labeled positive (entailment) and **hard_neg** is labeled negative (contradiction) because **sent0** and **sent1** share the same label (Offensive) in OLID, while **hard_neg** has the opposite label (Not Offensive). This new NLI dataset forces SimCSE to consider both semantic similarity and offensiveness when encoding sentences. The dataset contains 40,368 sentences, each with a positive (entailment) and a hard_neg (contradiction) pair.

3.5 Optimized SimCSE for data augmentation

SimCSE$_{\text{optimized}}$ is an enhanced version of SimCSE$_{\text{supervised}}$, fine-tuned on a new NLI dataset to better capture both semantic equivalence and offensiveness in sentences. Detailed in Sect. 3.4, this dataset includes labeled sentences that specify whether they are semantically equivalent and/or contain offensive content. The retraining process equips SimCSE$_{\text{optimized}}$ to discern subtle language nuances that differentiate between similar meanings and offensive contexts.

Upon retraining, SimCSE$_{\text{optimized}}$ generates sentence representations that reflect both semantic and offensive contexts. This dual consideration enables it to evaluate similarity between sentences more effectively, incorporating both their meanings and sentiments. Such capabilities are vital for tasks like cyberbullying detection, where it is crucial to identify offensive content that might be masked as humor or compliments.

By leveraging CSSL, SimCSE$_{\text{optimized}}$ enriches the training set with diverse and semantically similar data, enhancing the model's generalization ability. This approach allows for a more nuanced understanding of language, aiding in the detection of complex cyberbullying instances. To augment the data, the following steps were taken:

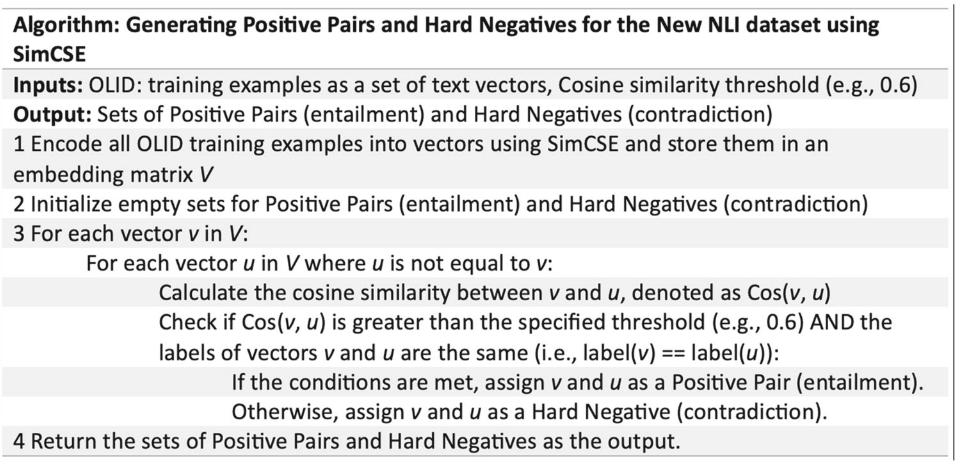
3.5.1 Training the parallel BERT + Bi-LSTM model

The proposed model begins by training the parallel BERT + Bi-LSTM model using the OLID dataset. It identifies misclassified examples (OLIDm), which are sentences incorrectly labeled as offensive or not offensive. Both types of misclassified sentences are used to augment and expand the training set.

3.5.2 Finding misclassified examples

Misclassified examples are those where the parallel BERT + Bi-LSTM model incorrectly labels offensive examples as not offensive, and vice versa, were used to expand the training set. By augmenting these examples, we aimed to identify the model's weak points and enhance its learnability, helping it learn from its mistakes and become more robust.

Misclassified examples were identified from the training split after implementing early stopping, a technique used to halt training to prevent overfitting and enhance model generalization. This method involves stopping training when performance on a validation set deteriorates, allowing for performance monitoring and model fine-tuning. Evaluations occur after each training epoch to assess whether the model



Algorithm: Generating Positive Pairs and Hard Negatives for the New NLI dataset using SimCSE

```
Inputs: OLID: training examples as a set of text vectors, Cosine similarity threshold (e.g., 0.6)
Output: Sets of Positive Pairs (entailment) and Hard Negatives (contradiction)
1 Encode all OLID training examples into vectors using SimCSE and store them in an
  embedding matrix V
2 Initialize empty sets for Positive Pairs (entailment) and Hard Negatives (contradiction)
3 For each vector v in V:
      For each vector u in V where u is not equal to v:
          Calculate the cosine similarity between v and u, denoted as Cos(v, u)
          Check if Cos(v, u) is greater than the specified threshold (e.g., 0.6) AND the
          labels of vectors v and u are the same (i.e., label(v) == label(u)):
              If the conditions are met, assign v and u as a Positive Pair (entailment).
              Otherwise, assign v and u as a Hard Negative (contradiction).
4 Return the sets of Positive Pairs and Hard Negatives as the output.
```

Fig. 4 Algorithm for creating the new NLI dataset

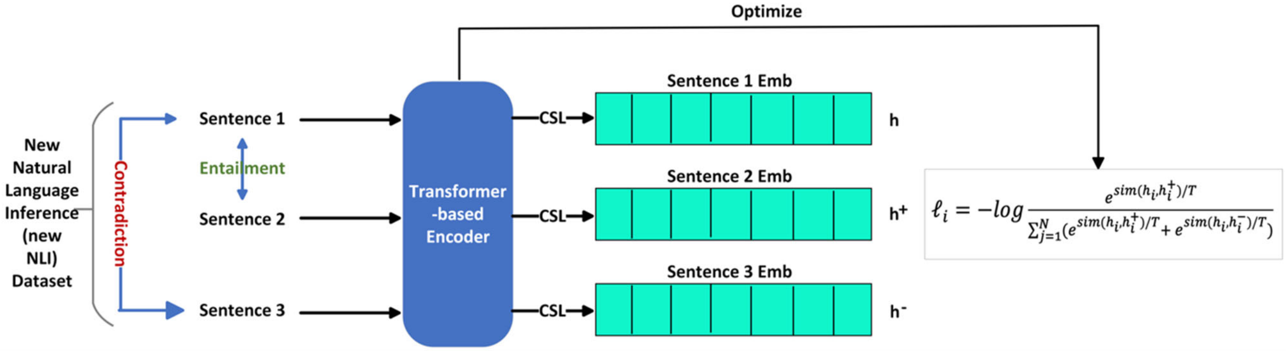
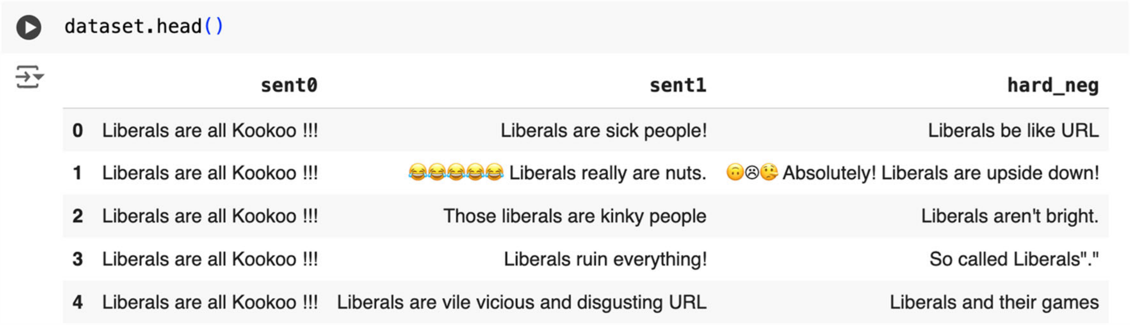

Optimize
New Natural Language Inference (new NLI) Dataset
Contradiction
Entailment
Sentence 1
Sentence 2
Sentence 3
Transformer -based Encoder
CSL
CSL
CSL
Sentence 1 Emb
Sentence 2 Emb
Sentence 3 Emb
h
h+
h-
$\ell_i = -log \frac{e^{sim(h_i,h_i^+)/T}}{\Sigma_{j=1}^N (e^{sim(h_i,h_i^+)/T} + e^{sim(h_i,h_i^-)/T})}$

Fig. 5 The process of retraining SimCSE$_{\text{supervised}}$ on the new NLI dataset



```
dataset.head()
```

	sent0	sent1	hard_neg
0	Liberals are all Kookoo !!!	Liberals are sick people!	Liberals be like URL
1	Liberals are all Kookoo !!!	😂😂😂😂😂 Liberals really are nuts.	🙃☹🤥 Absolutely! Liberals are upside down!
2	Liberals are all Kookoo !!!	Those liberals are kinky people	Liberals aren't bright.
3	Liberals are all Kookoo !!!	Liberals ruin everything!	So called Liberals"."
4	Liberals are all Kookoo !!!	Liberals are vile vicious and disgusting URL	Liberals and their games

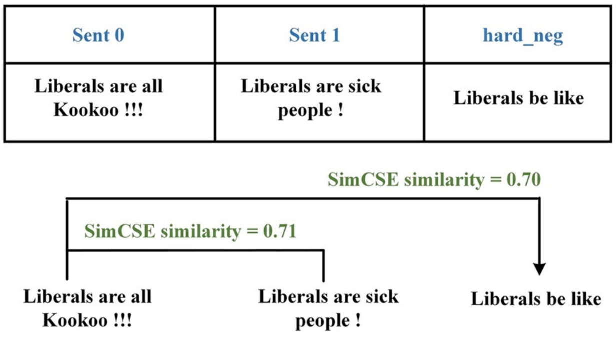
Fig. 6 Screenshot of the new NLI dataset

is improving, overfitting, or underfitting. By keeping non-trained examples in the training set, considered as complex and challenging, early stopping aids in reducing overfitting. Consequently, these misclassified examples were carefully augmented to minimize noise and emphasize complex patterns.

3.5.3 Using SimCSE$_{\text{optimized}}$ for encoding

The OLID$_{\text{m}}$ and SOLID training sets will be the inputs to the SimCSE$_{\text{optimized}}$ model as M and S, respectively. SimCSE$_{\text{optimized}}$ encodes M and S, and generates new representations for both datasets, as M_h and S_h, respectively.

- $M_h = \text{simcse.encode}\ (\text{OLID}_{\text{m}})\ \{M \in R^{\text{w}*n}\}$
- $S_h = \text{simcse.encode}\ (\text{crawled\_text}\ (\text{SOLID}))\ \{S \in R^{m*n}\}$



Sent 0	Sent 1	hard_neg
Liberals are all Kookoo !!!	Liberals are sick people !	Liberals be like

Fig. 7 Similarities between sentences in the new NLI dataset

Where M represents the original representations of OLID_{m} and M_h its new representations generated by $\text{SimCSE}_{\text{optimized}}$, and S represents the original SOLID training set and S_h its new representations generated by $\text{SimCSE}_{\text{optimized}}$. The $\text{SimCSE}_{\text{optimized}}$ model will use the new representations M_{h} and S_{h} to calculate their similarity scores and find a group of similar examples for M_{h} from S_{h}.

3.5.4 Measuring cosine similarity between examples

$\text{SimCSE}_{\text{optimized}}$ is integrated to calculate the cosine similarities between the representations in M_h and S_h, taking into account both the semantic and offensive meaning. Each sentence in M has a group of candidate sentences from S which achieve cosine similarity greater than the threshold of 0.7.

The $\text{SimCSE}_{\text{optimzed}}$ model calculates the cosine similarity using Eq. (2) between each representation h_{oi} in M_h $\{h_{oi} \in M_h,\ 1 \leq i \leq k\}$ and each representation h_{si} in S_h $\{h_{si} \in S_h,\ 1 \leq i \leq m\}$ and creates a group of similarities for each h_{oi}.

$$\cos(\theta) = \frac{S_h \cdot M_h}{|S_h||M_h|} \tag{2}$$

3.5.5 Assigning the correct *pseudo* label using label propagation (LP)

The Label Propagation (LP) algorithm is a semi-supervised learning method that improves on supervised techniques by leveraging both labeled and unlabeled data. It constructs a graph $G = (V,E)$ as shown in Fig. 8, where V represents labeled (L) and unlabeled (U) examples, and edges E represent the similarity between nodes i and j with weight wij. Weights wij are higher for nodes that are closer (more similar).

Based on the consistent assumption that nearby nodes (sentence embeddings) are likely to have the same label, we can perform LP to propagate information from samples with known labels to samples without labels or with noisy labels.

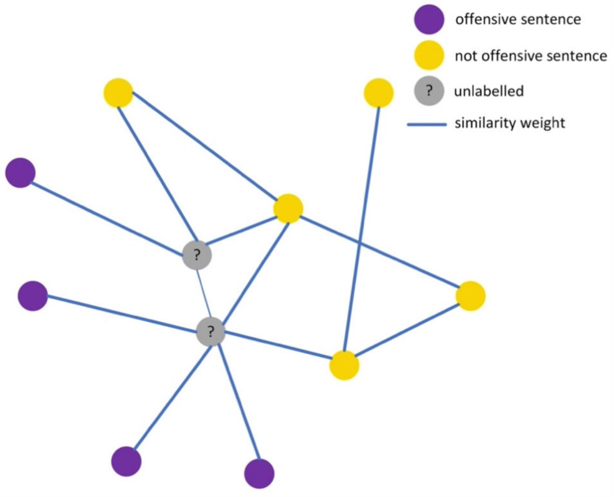


Fig. 8 LP graph

LP is used in the proposed model to assign the correct labels to the candidate examples for each h_{oi} resulting from data augmentation by calculating cosine similarities. The LP algorithm is trained on the OLID training set (after encoding OLID training set using $\text{SimCSE}_{\text{optimised}}$) and used to create a graph, as shown in Fig. 8, using the candidate examples from SOLID and the OLID training set (after encoding them using $\text{SimCSE}_{\text{optimised}}$) and conducts the cosine similarity to calculate the distance metric as demonstrated in Eq. (3).

$$1 - \frac{u.V}{||u||||V||} or 1 - \cos(\theta) \tag{3}$$

The LP algorithm uses Eq. (3) to calculate distances between labeled and unlabeled examples based on cosine similarity. The greater the similarity between two nodes, the shorter the distance between them. Strict criteria ensure high validity by requiring that the original label matches LP's label exactly. Approved candidate examples are added to the OLID training set, creating the $\text{OLID}_{\text{augmented}}$ dataset for training the cyberbullying detection model.

Figure 9 illustrates the data augmentation process, which includes training the parallel BERT + Bi-LSTM on OLID, identifying misclassified examples, encoding these and the SOLID dataset with $\text{SimCSE}_{\text{optimized}}$, calculating cosine similarities, identifying candidate examples, and assigning pseudo labels using the LP algorithm. If the probability of the pseudo label for the candidate example (calculated by LP) is greater than or equal to the predefined threshold ($> =$ 0.8) and matches the label of its similar example in the misclassified examples, this candidate example will be added to the augmented data; otherwise, the candidate example will be ignored.

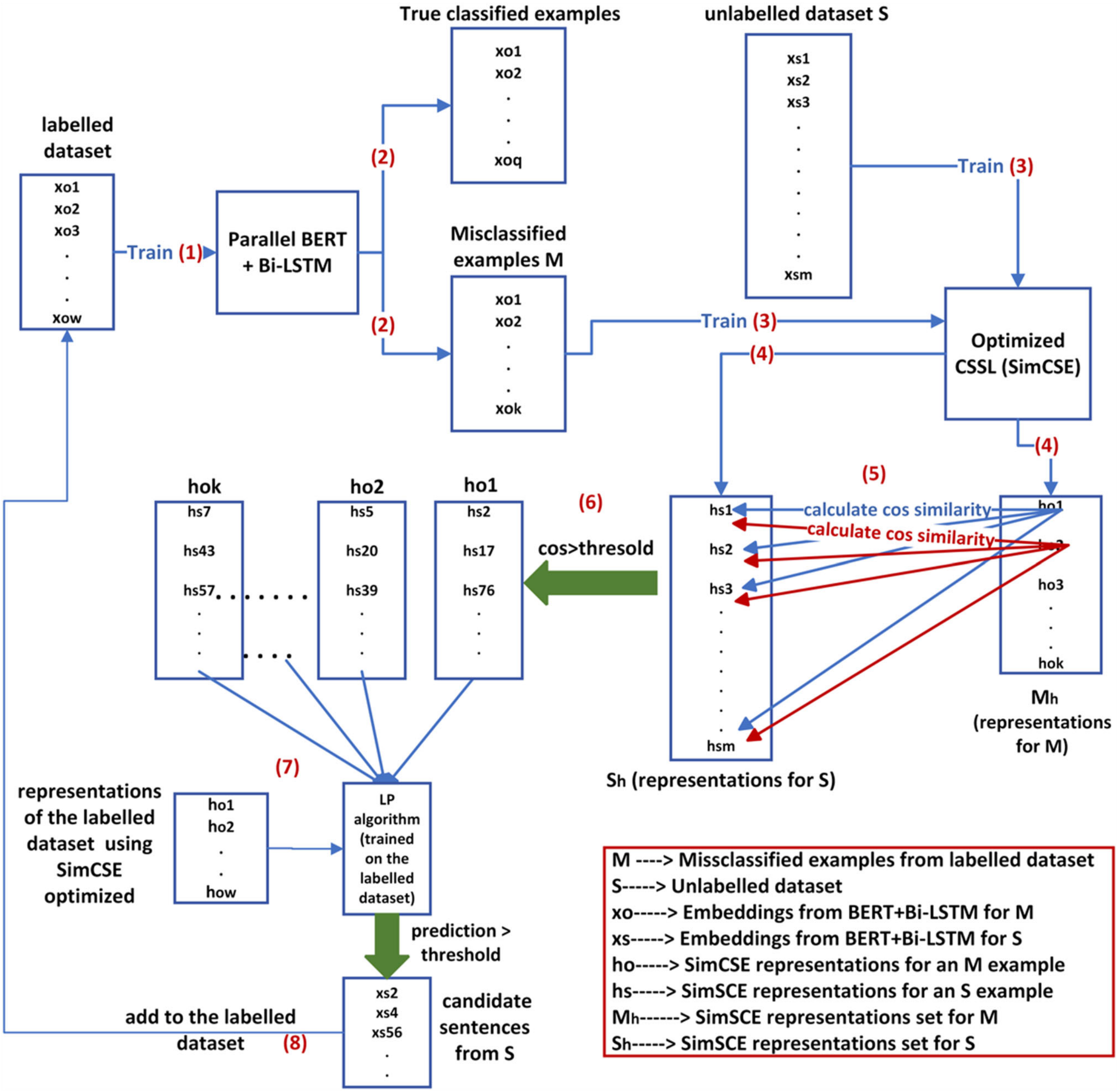

True classified examples
xo1
xo2
xoq
unlabelled dataset S
xs1
xs2
xs3
xsm
labelled dataset
xo1
xo2
xo3
xow
Train (1)
Parallel BERT + Bi-LSTM
(2)
(2)
Misclassified examples M
xo1
xo2
xok
Train (3)
Train (3)
(4)
(4)
Optimized CSSL (SimCSE)
(5)
(6)
hok
hs7
hs43
hs57
ho2
hs5
hs20
hs39
ho1
hs2
hs17
hs76
cos>thresold
hs1
hs2
hs3
hsm
calculate cos similarity
calculate cos similarity
ho1
ho2
ho3
hok
Mh (representations for M)
Sh (representations for S)
(7)
representations of the labelled dataset using SimCSE optimized
ho1
ho2
how
LP algorithm (trained on the labelled dataset)
prediction > threshold
xs2
xs4
xs56
candidate sentences from S
add to the labelled dataset
(8)
M ----> Missclassified examples from labelled dataset
S-----> Unlabelled dataset
xo-----> Embeddings from BERT+Bi-LSTM for M
xs-----> Embeddings from BERT+Bi-LSTM for S
ho-----> SimCSE representations for an M example
hs-----> SimSCE representations for an S example
Mh------> SimSCE representations set for M
Sh-----> SimSCE representations set for S

Fig. 9 Data augmentation for the proposed model

3.5.6 Detecting cyberbullying

Our proposed model leverages the parallel BERT + Bi-LSTM architecture, detailed in one of our publications [11], trained from on the OLID$_{\text{augmented}}$ dataset for cyberbullying detection. This dataset expands training data using large-scale unlabeled data for increased diversity and generalizability.

Training on the OLID$_{\text{augmented}}$ dataset introduces a broader variety of text examples, improving the model's ability to capture the nuances of cyberbullying language and reducing overfitting by leveraging a more recent and diverse dataset [35].

Figure 3 shows the process where each sentence is input into both BERT and Bi-LSTM, creating distinct representations. BERT processes the sentence directly, while Bi-LSTM uses GloVe embeddings. The CLS token and Bi-LSTM outputs are fed into two fully connected neural networks (FCNNs). This process is discussed in Sect. 3.2.

The testing was carried out using the SOLID testing set, one of the competition datasets, which adheres to the rules of the SemEval-2020 Task 12 (OffensEval 2020) competition.

This testing set has been manually annotated and serves as the benchmark for evaluating the performance of participants' models.

4 Results analysis

4.1 Implementation details

BERT has developed with two principal variants: $\text{BERT}_{\text{base}}$, which possesses 110 million parameters, and $\text{BERT}_{\text{large}}$, which is significantly larger, containing 336 million parameters—exactly three times the size of $\text{BERT}_{\text{base}}$. In terms of structure, $\text{BERT}_{\text{base}}$ is equipped with 12 layers, while $\text{BERT}_{\text{large}}$ boasts 24. This difference implies that running experiments on $\text{BERT}_{\text{large}}$ could take up to three times as long as those on $\text{BERT}_{\text{base}}$. Both models are built with feed-forward networks, but they differ in capacity: $\text{BERT}_{\text{base}}$ has 768 hidden units per layer and 12 attention heads, whereas $\text{BERT}_{\text{large}}$ has 1024 hidden units and 16 attention heads.

We used Google Colab with one Graphic Processing Unit (GPU), which limits work on huge models such as $\text{BERT}_{\text{large}}$. Therefore, to facilitate experimentation we used $\text{BERT}_{\text{base}}$, which is made up of 12 layers and 512 tokens, with an embedding length of 768 for each token.

In BERT's FCNN, the Rectified Linear Unit (ReLU) serves as the activation function for all intermediate layers, with the exception of the output layer, which utilizes SoftMax. The SoftMax function is widely adopted in the output layers of neural networks to compute a multinomial probability distribution. Conversely, the Bi-LSTM architecture employs the hyperbolic tangent (Tanh) function in its intermediate layers.

We used Grid Search, a process that builds and evaluates a model for each combination of parameters to determine the best filter-hyperparameter. Grid Search used the following ranges:{"epochs":[2, 6], "Learning_rate":[1e-5,1e-6], "weight_decay": [0.1,0.001], "batch_size":[16, 128]}. Parallel BERT + Bi-LSTM is fine-tuned with a learning rate of 9e-06 for 7 epochs with a batch size of 64 and 0.01 weight_decay. The hyperparameters that were used to fine-tune $\text{SimCSE}_{\text{optimized}}$ are batch size 64, number of epochs 5, and learning rate 3e-5.

4.2 Evaluation metrics

For this research, we have used two standard metrics for evaluating the performance of the proposed model developed: F1-score and accuracy. F1-score is a symmetric single-value representation of a model's precision and recall derived from the harmonic mean of the precision and recall values. Following the SemEval 2020 – (Offenseval 2020) competition organizers' request, the macro F1-score was used as the primary metric for all sub-tasks. Another reason for using the F1-score was that it offers a balanced measure of a model's performance, taking into account both precision (the accuracy of identified instances) and recall (the ability to identify all actual instances). Furthermore, this evaluation metric was employed to facilitate a fair comparison between the proposed model and the top three models in the competition.

For tasks like offensive language detection, especially when using datasets like OLID and SOLID, the F1-score is generally more appropriate than accuracy for several reasons:

(1) *Class imbalance*: Offensive language datasets often have an imbalanced class distribution, where the majority of instances are non-offensive, and a smaller proportion are offensive. In such cases, the accuracy can be misleading. For example, a dataset with 99% non-offensive and 1% offensive examples could have a model that predicts "non-offensive" for every example and achieves an accuracy of 99%. However, this model would be completely ineffective in identifying offensive language.

(2) *Cost of errors*: In offensive language detection, the cost of false positives and false negatives could vary significantly. A false negative, which is failing to identify an offensive comment, could have more severe consequences compared to a false positive, which is incorrectly flagging a non-offensive comment. To evaluate the performance of a model in terms of both precision and recall, the F1-score provides a balanced measure.

(3) *Focus on positive class*: In binary classification problems like offensive language detection, the primary interest is often in the performance of the model on the positive (minority) class, not the overall accuracy. F1-score provides a measure that focuses more on the performance concerning the positive class.

Considering the nature of the task and the characteristics of the OLID and SOLID datasets, the F1-score is generally the more appropriate metric for evaluating models in offensive language detection. It offers a more nuanced and informative measure of a model's ability to correctly identify offensive language, especially when dealing with imbalanced classes or when the costs of different types of errors are not equal.

Accuracy indicates the ratio of correct predictions, including both 'OFF' and 'NOT' categories, to the total number of predictions. We have recorded the levels of accuracy for both training and validation phases based on the data from the latest training epoch. Although accuracy is a less reliable metric in the presence of imbalanced datasets, it offers an overview of the model's overall performance.

4.3 Validation

To validate our proposed model, we conducted different experiments and ablation studies to investigate the impact of each component or feature on the performance of the proposed model as follows:

1- Evaluate the annotation performance of various SimCSE variations using LP algorithm, including $\text{SimCSE}_{\text{optimized}}$, $\text{SimCSE}_{\text{supervised}}$, and $\text{SimCSE}_{\text{unsupervised}}$ (SubSect. 4.3.1, Table 2);
2- Evaluate the annotation performance of CSSL approach using $\text{SimCSE}_{\text{optimized}}$ and $\text{SimCSE}_{\text{supervised}}$ (SubSect. 4.3.1, Table 3);
3- Assess the efficiency of using $\text{SimCSE}_{\text{optimized}}$ for data augmentation on the performance of the proposed model for cyberbullying detection by comparing the performance of the proposed model using augmented data from other variations of the SimCSE model (SubSect. 4.3.2, Table 4);
4- Investigate the impact of adding augmented data to the training set on the performance of the proposed model to evaluate the impact of data augmentation on model performance. For this purpose, the model was trained twice: once on the OLID training set and once on the augmented data $\text{OLID}_{\text{augmented}}$ (SubSect. 4.3.3, Table 5);
5- Compare the performance of the proposed model for cyberbullying detection with baseline models (SubSect. 4.3.4, Tables 6 and 7);
6- Evaluate the generalizability of the proposed model by testing it on different datasets, such as Yelp (SubSect. 4.3.5, Table 8).

The performance of the proposed model parallel BERT + Bi-LSTM using $\text{SimCSE}_{\text{optimized}}$ for augmented data was compared for detecting cyberbullying with the top three teams for English subtask A in the competition SemEval-2020 Task 12: Multilingual Offensive Language Identification in Social Media (Offens Eval 2020) [34] using F1-scores. In addition, the performance of the proposed model was compared with benchmark models such as BERT and RoBERTa. Comparing the proposed model with the top three teams from SemEval-2020 Task 12 and benchmark models like BERT and RoBERTa serves several purposes:

(1) By measuring our model against the top models from the SemEval-2020 Task 12, we can directly assess its performance relative to the highest achieved standards in this specific domain. Such a comparison is crucial for situating our work within the cutting edge of research advancements. In addition, we employed the same datasets and evaluation metrics used in SemEval-2020 Task 12 to ensure the fairness and validity of our comparison.
(2) BERT and RoBERTa have set the groundwork for numerous advancements in natural language processing. These models act as benchmarks that are both reputable and widely adopted, thus serving as a baseline for evaluating the improvements offered by our model. Demonstrating superior performance against these benchmarks can substantiate the significance of our model's contributions.
(3) By benchmarking against both specialized competition models and versatile pre-trained models, we seek to demonstrate that our approach is not only competitive in a high-stakes setting but also robust across general applications. This breadth of comparison assures the community of our model's applicability to varied datasets and contexts

4.3.1 The performance of $\text{SimCSE}_{\text{optimized}}$ with LP for data annotation

Three experiments were conducted to validate three different variations of SimCSE: $\text{SimCSE}_{\text{supervised}}$, $\text{SimCSE}_{\text{unsupervised}}$, and $\text{SimCSE}_{\text{optimized}}$ for data annotation. We repeated the following steps for each SimCSE version:

1. Using the SimCSE version to encode the OLID training set and convert it to embeddings;
2. Removing labels from the SOLID test set and encoding it using the SimCSE version;
3. Using LP to create the graph and calculating the distance between the embeddings using Eq. (1);
4. Assigning labels for the SOLID test set using LP;

Comparing the results of labeling the SOLID test from LP with the true labels.

Figure 10 demonstrates the algorithm of using the three SimCSE variations for data annotation using the LP algorithm for assigning the pseudo labels.

Table 2 shows a comparison between the three versions of SimCSE—$\text{SimCSE}_{\text{supervised}}$, $\text{SimCSE}_{\text{unsupervised}}$, and $\text{SimCSE}_{\text{optimized}}$ when used in conjunction with the LP algorithm for data annotation. These versions are evaluated using the OLID + SOLID datasets for the task of data annotation in the context of cyberbullying detection.

As shown in Table 2, all three models of SimCSE have relatively high F1-scores and accuracy, indicating that they are generally effective for the task of data annotation. However, the $\text{SimCSE}_{\text{optimized}}$ Embeddings with LP method received the highest F1-score (0.8909), demonstrating its efficacy in data annotation for the detection of cyberbullying.

Table 2 Validation of SimCSE versions for data annotation with LP using OLID training set and SOLID test set

Model	F1-score	Precision	Recall	Accuracy
LP With SimCSE$_{\text{supervised}}$ embeddings	0.8076	0.8082	0.8069	0.8459
LP With SimCSE$_{\text{unsupervised}}$ embeddings	0.8002	0.8051	0.7967	0.8091
LP With SimCSE$_{\text{optimized}}$ embeddings	**0.8909**	**0.8822**	**0.9014**	**0.9102**

Table 3 Validation of using CSSL approach for data annotation using the OLID training set and SOLID test set

Model	Dataset	F1-score	Precision	Recall	Accuracy
SimCSE$_{\text{supervised}}$	OLID training set + SOLID test	0.70	0.69	0.72	0.74
SimCSE$_{\text{optimized}}$		**0.81**	**0.80**	**0.83**	**0.84**

Table 4 Results of the proposed model using three versions of SimCSE

Model	Augmentation model	Number of augmentations	F1-score	Precision	Recall	Accuracy
Parallel BERT + Bi-LSTM	SimCSE$_{\text{supervised}}$	6,339	0.9164	0.8919	0.9400	0.9297
Parallel BERT + Bi-LSTM	SimCSE$_{\text{unsupervised}}$	2,420	0.9197	0.9031	0.9444	0.9320
Parallel BERT + Bi-LSTM	SimCSE$_{\text{optimized}}$	**5,914**	**0.9231**	**0.9046**	**0.9475**	**0.9349**

Table 5 The performance of the proposed model trained on OLID and OLID$_{\text{augmented}}$

Model	No. of augmentations	F1-score	Precision	Recall	Accuracy	Dataset
Parallel BERT + Bi-LSTM with SimCSE$_{\text{optimized}}$	–	0.9156	0.8976	0.9444	0.9279	OLID
	5,914	**0.9231**	**0.9046**	**0.9475**	**0.9349**	**OLID$_{\text{augmented}}$**

Table 6 Results of the proposed model with the top three model in OffensEval-2020 competition

Model	F1-score
UHH-LT	0.9204
Galileo	0.9198
Rouges	0.9187
Parallel BERT + Bi-LSTM with SimCSE$_{\text{optimized}}$ for augmented data	**0.9231**

Table 7 Results of the proposed model with BERT and RoBERTa

Model	Augmentation model	Number of augmentations	F1-score	Precision	Recall	Accuracy
BERT	SimCSE$_{\text{optimized}}$	16,571	0.9194	0.9014	0.9481	0.9313
RoBERTa		5,291	0.9163	0.8975	0.9483	0.9282
Parallel BERT + Bi-LSTM Model		5,914	**0.9231**	**0.9046**	**0.9475**	**0.9349**

Table 8 Results of the proposed model trained on the yelp dataset

Model	Augmentation model	Number of augmentations	F1-score	Precision	Recall	Accuracy
Parallel BERT + Bi-LSTM Model	SimCSE$_{\text{optimized}}$	2,726	0.9523	0.9458	0.9595	0.9593
BERT		1,447	0.9493	0.9435	0.9558	0.9568
RoBERTa		487	0.9680	0.9647	0.9716	0.9730

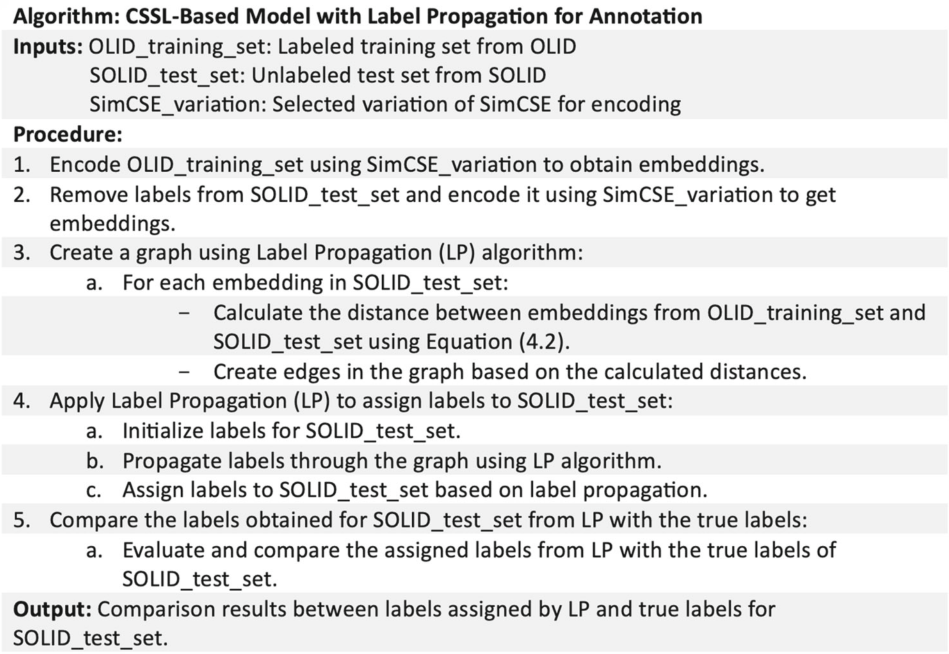


Fig. 10 Algorithm for using CSSL-Based model with Label Propagation for Annotation

```
Algorithm: CSSL-Based Model with Label Propagation for Annotation
Inputs: OLID_training_set: Labeled training set from OLID
        SOLID_test_set: Unlabeled test set from SOLID
        SimCSE_variation: Selected variation of SimCSE for encoding
Procedure:
1. Encode OLID_training_set using SimCSE_variation to obtain embeddings.
2. Remove labels from SOLID_test_set and encode it using SimCSE_variation to get
   embeddings.
3. Create a graph using Label Propagation (LP) algorithm:
      a. For each embedding in SOLID_test_set:
            – Calculate the distance between embeddings from OLID_training_set and
              SOLID_test_set using Equation (4.2).
            – Create edges in the graph based on the calculated distances.
4. Apply Label Propagation (LP) to assign labels to SOLID_test_set:
      a. Initialize labels for SOLID_test_set.
      b. Propagate labels through the graph using LP algorithm.
      c. Assign labels to SOLID_test_set based on label propagation.
5. Compare the labels obtained for SOLID_test_set from LP with the true labels:
      a. Evaluate and compare the assigned labels from LP with the true labels of
         SOLID_test_set.
Output: Comparison results between labels assigned by LP and true labels for
SOLID_test_set.
```

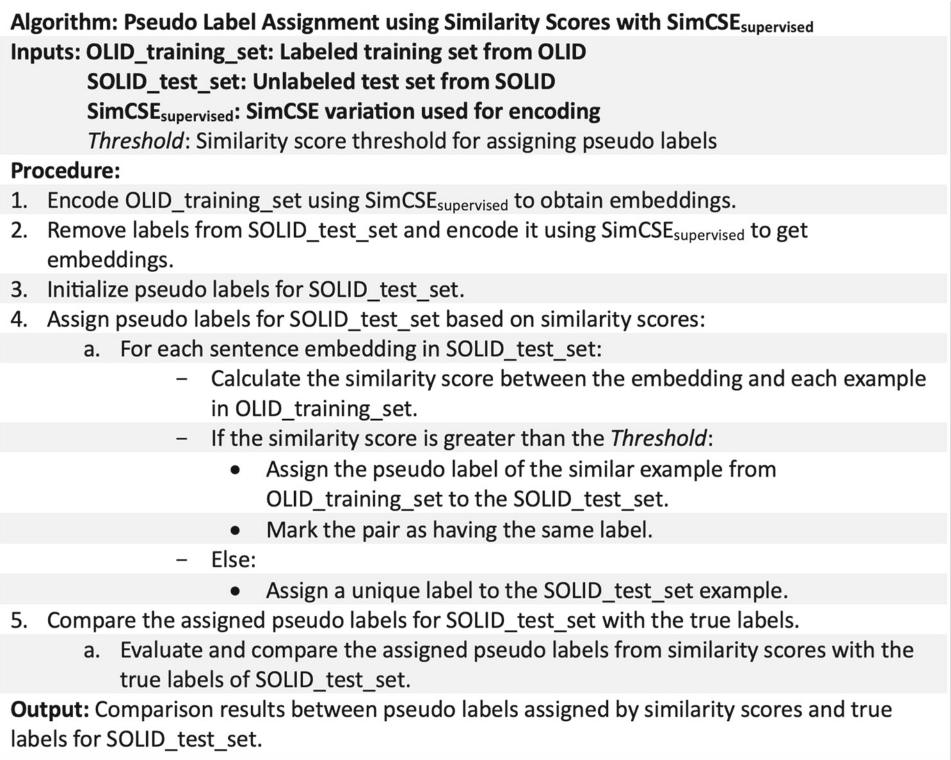
While still producing acceptable results, SimCSE$_{\text{supervised}}$ and SimCSE$_{\text{unsupervised}}$ lagged behind in overall performance.

In addition, another experiment was done to explore the performance of using the CSSL approach for data annotation by using SimCSE$_{\text{supervised}}$ and SimCSE$_{\text{optimized}}$ to encode the examples and calculate the cosine similarities without using LP for assigning the pseudo label. The similarity score generated by SimCSE was employed to assign the pseudo labels to each example. The CSSL approach represented by SimCSE$_{\text{supervised}}$ and SimCSE$_{\text{optimized}}$ was validated for data annotation. For SimCSE$_{\text{supervised}}$, the following steps were conducted:

1. Using SimCSE$_{\text{supervised}}$ to encode the OLID training set and convert it to embeddings;
2. Removing labels from the SOLID test set and encoding it using the SimCSE$_{\text{supervised}}$;
3. Assigning labels for the SOLID test set based on the similarity scores;
4. If the similarity score between the sentences is greater than the threshold, the two examples will have the same label similar to the labelled example;
5. Comparing the results of the labels for the SOLID test result from calculating the similarity with the true labels (the labels annotated by humans).

Figure 11 demonstrates the algorithm of using SimCSE$_{\text{supervised}}$ for data annotation using the OLID training set and SOLID test set. The same algorithm was repeated using SimCSE$_{\text{optimized}}$. Table 3 shows the performance using F1-scores and accuracy for using the CSSL approach for data annotation. The performance of SimCSE$_{\text{optimized}}$ is greater than that of SimCSE$_{\text{supervised}}$ for the OLID training set + SOLID test set as it achieved better F1-scores and accuracy. This is because SimCSE$_{\text{optimized}}$ is optimized and fine-tuned by retraining it on the new NLI dataset to force the model to take into account the underlying meaning including the semantic and offensive meanings.

Based on the results provided in Table 3, it's clear that SimCSE$_{\text{optimized}}$ outperforms SimCSE$_{\text{supervised}}$ for data annotation on the OLID dataset across all metrics (F1-score,



```
Algorithm: Pseudo Label Assignment using Similarity Scores with SimCSE_supervised
Inputs: OLID_training_set: Labeled training set from OLID
        SOLID_test_set: Unlabeled test set from SOLID
        SimCSE_supervised: SimCSE variation used for encoding
        Threshold: Similarity score threshold for assigning pseudo labels
Procedure:
1. Encode OLID_training_set using SimCSE_supervised to obtain embeddings.
2. Remove labels from SOLID_test_set and encode it using SimCSE_supervised to get
   embeddings.
3. Initialize pseudo labels for SOLID_test_set.
4. Assign pseudo labels for SOLID_test_set based on similarity scores:
     a. For each sentence embedding in SOLID_test_set:
          – Calculate the similarity score between the embedding and each example
            in OLID_training_set.
          – If the similarity score is greater than the Threshold:
               • Assign the pseudo label of the similar example from
                 OLID_training_set to the SOLID_test_set.
               • Mark the pair as having the same label.
          – Else:
               • Assign a unique label to the SOLID_test_set example.
5. Compare the assigned pseudo labels for SOLID_test_set with the true labels.
     a. Evaluate and compare the assigned pseudo labels from similarity scores with the
        true labels of SOLID_test_set.
Output: Comparison results between pseudo labels assigned by similarity scores and true
labels for SOLID_test_set.
```

Fig. 11 Algorithm for pseudo label assignment using similarity scores with $\text{SimCSE}_{\text{supervised}}$

precision, recall, and accuracy), suggesting that retraining $\text{SimCSE}_{\text{supervised}}$ on the new NLI dataset created using OLID data has improved the model's ability to generalize and make accurate predictions.

As discussed before, SimCSE is not used for classification tasks as it is used to generate augmented data, thus another algorithm for choosing the offensive candidate augmented examples was used. Using another algorithm with $\text{SimCSE}_{\text{supervised}}$ for assigning the pseudo labels after calculating the similarities, increases the performance for the labeling, thus improving the detection performance for cyberbullying.

Furthermore, the LP algorithm was employed to assign pseudo labels after calculating similarities using various SimCSE versions. This method improved performance in data annotation when using OLID training set as presented in Table 2.

4.3.2 Comparisons of $\text{SimCSE}_{\text{optimized}}$ for cyberbullying detection with SimCSE versions

The proposed model was trained three times for detecting offensive words using three different versions—$\text{SimCSE}_{\text{supervised}}$, $\text{SimCSE}_{\text{unsupervised}}$, and $\text{SimCSE}_{\text{optimized}}$ to validate the performance of the proposed model for cyberbullying detection using the augmented data results from $\text{SimCSE}_{\text{optimized.}}$ The parallel BERT + Bi-LSTM achieved 0.9164, 0.9197, and 0.9231, respectively, for English subtask A. As shown in Table 4, the number of augmentations differs for each SimCSE version, but using $\text{SimCSE}_{\text{optimized}}$ to generate the augmented data achieves the highest performance on cyberbullying detection.

In this experiment, the aim is to validate the performance of the proposed model parallel BERT + Bi-LSTM for cyberbullying detection using augmented data, particularly focusing on the detection of offensive words. To this end, the model is trained three times using three different augmented datasets generated using three different versions of SimCSE—$\text{SimCSE}_{\text{supervised}}$, $\text{SimCSE}_{\text{unsupervised}}$, and $\text{SimCSE}_{\text{optimized}}$. The results were quite revealing, especially in terms of F1-scores achieved for English subtask A.

As shown in Table 4, although the number of augmentations varied across the different versions of SimCSE, the model trained with $\text{SimCSE}_{\text{optimized}}$ yielded the highest performance in terms of both F1-score and accuracy. This suggests that $\text{SimCSE}_{\text{optimized}}$ is the most effective version for data augmentation in the context of cyberbullying detection based on the metrics considered in this work.

A noticeable improvement in performance using the parallel BERT + Bi-LSTM model is seen moving from $\text{SimCSE}_{\text{supervised}}$ to $\text{SimCSE}_{\text{optimized}}$, with $\text{SimCSE}_{\text{unsupervised}}$ falling in between. This suggests that the optimization process in SimCSE$_{\text{optimized}}$ enhances the quality of embeddings and annotations, resulting in better model performance. The F1-score consistently increases, highlighting the impact of fine-tuning the SimCSE model.

It's worth noting that the number of the augmented data varies among these configurations. Parallel BERT + Bi-LSTM trained on the augmented data generated by SimCSE$_{\text{optimized}}$, which achieved the highest F1-score, utilized moderate augmented data (5,914). This suggests that achieving optimal performance does not necessarily require excessive augmented data, emphasizing the importance of efficient data augmentation strategies.

In conclusion, the results in Table 4 highlight the effectiveness of different SimCSE variations on generating the augmented data. The best-performing option is SimCSE$_{\text{optimized}}$ with LP, which achieves a better F1-score while keeping a balanced trade-off between recall and precision. These findings highlight the significance of optimization strategies in CSSL-based models and their potential to improve cyberbullying detection performance.

4.3.3 The impact of the augmented data on cyberbullying detection

To evaluate the impact of the augmented data on the detection performance, the model parallel BERT + Bi-LSTM was trained twice for detecting offensive words. One training was done using only OLID, and the other was done using OLID$_{\text{augmented}}$ data generated by SimCSE$_{\text{optimized}}$. Our main focus was on subtask A, which deals with language detection in English. English subtask A is a part of the SemEval 2020 – Task 12 competition, which aims to classify tweets in English language into OFF (offensive tweet) or NOT (not offensive tweet). To accomplish this, we utilized a combination of parallel BERT and Bi LSTM models with the augmented data generated by SimCSE$_{\text{optimized}}$.

The results in Table 5 show that for English subtask A, the proposed model parallel BERT + Bi-LSTM achieved a macro average F1-score 0. 9156 when trained on the OLID training set only. On the other hand, parallel BERT + Bi-LSTM achieved 0.9231 when trained on OLID$_{\text{augmented}}$ data generated using SimCSE$_{\text{optimized}}$.

The performance increased when trained on OLID$_{\text{augmented}}$ by more than 0.5% for F1-score compared to training it on OLID only.

The dataset used greatly influences the performance of the model parallel BERT + Bi-LSTM for detecting cyberbullying. The OLID$_{\text{augmented}}$ dataset, which benefits from data augmentation based on SimCSE, offers an extensive and varied training set compared to OLID alone. This diversity seems to have an impact on the model's capacity to generalize and accurately identify offensive language.

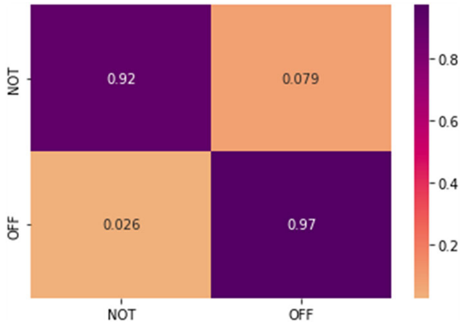


Fig. 12 The confusion matrix for the proposed model using SimCSE$_{\text{optimized}}$

4.3.4 Comparing the proposed model with baseline models

To validate the proposed model parallel BERT + Bi-LSTM, different experiments were conducted using different variations of SimCSE, different baseline models, and different datasets. The performance of the proposed model parallel BERT + Bi-LSTM using SimCSE$_{\text{optimized}}$ for augmented data was compared for detecting cyberbullying with the top three teams for English subtask A in the competition SemEval-2020 Task 12: Multilingual Offensive Language Identification in Social Media (OffensEval 2020) [34] using F1-scores. The top three teams which achieved the highest F1-scores of 0.9204, 0.9198 and 0.9187 are UHH-LT [36], Galileo [37], and Rouges [38], respectively. Table 6 shows the comparison between the proposed model, which uses SimCSE$_{\text{supervised}}$ and SimCSE$_{\text{optimized}}$ respectively for data augmentation, with the baseline models. As shown in Table 6, the proposed model outperformed the top three models in the SemEval-2020 (OffensEval 2020) competition regardless of using SimCSE$_{\text{supervised}}$ or SimCSE$_{\text{optimized}}$.

Table 6 includes only the F1-scores of all models for comparison because the authors of the top three models in the OffensEval-2020 competition did not provide the results for precision, recall and accuracy.

Figure 12 presents the confusion matrix for the proposed parallel BERT + Bi-LSTM model when trained on the augmented data generated by SimCSE$_{\text{optimized}}$. It demonstrates the percentages of true negatives (TN) (top-left purple), false negatives (FN) (bottom-left peach), false positives (FP) (top-right peach), and true positives (TP) (bottom-right purple). FP happens when a model incorrectly labels a non-offensive comment as offensive, which can be inconvenient or cause reputational damage to the user who commented. On the other hand, FN happens when a model fails to identify an actual offensive comment, potentially allowing hate speech, harassment, or other forms of offensive language to go unchecked, causing harm to individuals or groups. In this case, the cost can be much higher.

We can see that the proportion of FN and FP are 2.6% and 7.9%, which are small. As we can see from the figure, the percentage of FP is greater than the percentage of the FN, which means that there are some examples that have the (NOT) label that are classified as (OFF). These examples might have words that are considered offensive, but the context of the sentence suggested that the meaning was not offensive.

In addition, the performance of the proposed parallel BERT + Bi-LSTM model was compared with other baseline models such as BERT and RoBERTa. The results of the comparisons between the proposed model parallel BERT + Bi-LSTM with SimCSE$_{\text{optimized}}$ for data augmentation are demonstrated in Table 7.

Table 7 presents the performance of three different models, BERT, RoBERTa, and Parallel BERT + Bi-LSTM, when trained on OLID$_{\text{augmented}}$ and OLID'$_{\text{augmented}}$ data for cyberbullying detection.

In summary, parallel BERT + Bi-LSTM consistently outperformed both BERT and RoBERTa in terms of F1-score and accuracy, indicating the effectiveness of combining BERT and Bi-LSTM for cyberbullying detection. Additionally, models utilizing SimCSE$_{\text{optimized}}$ generally achieved higher F1-scores than their SimCSE$_{\text{supervised}}$ counterparts, demonstrating the benefits of optimizing SimCSE for data augmentation. The choice of model depends on the specific trade-offs between precision and recall that are acceptable for the task at hand, considering the potential consequences of false positives and false negatives in cyberbullying detection.

4.3.5 The performance of the proposed model with different datasets

A search for additional datasets was conducted to expand the scope of comparisons and experiments, thereby strengthening the support for the effectiveness and validation of the proposed model. Despite extensive efforts, unfortunately a large-scale dataset specializing in cyberbullying or offensive language that can be divided into two parts, a small labelled part for seeding and a large unlabelled part for augmented data extraction, has not been identified. Because of this, alternative datasets were explored in other fields, such as the Yelp Dataset from, [39] a large-scale dataset specializing in restaurant reviews, to conduct experiments and comparisons. To replicate the OLID use-case, a 0.05 of the Yelp Polarity training datasets was used for training, and the remaining training data was used as an unlabelled corpus for augmentation. This resulted in a training set of 25,000 examples, a development set of 2,800 examples, and an unbalanced test set of 25,000 examples. Table 8 showcases the performance of our proposed model parallel BERT + Bi-LSTM with SimCSE$_{\text{optimized}}$ for generating augmented data using the Yelp Polarity dataset. The model achieved macro-averaged F1-scores of 0.9523 when using SimCSE$_{\text{optimized}}$ for data augmentation. These results also validate the effectiveness of our model on yelp dataset. Our model obtained competitive results, although in this case it was superseded by ROBERTa.

4.3.6 Computational cost of the proposed model

BERT is a transformer-based deep learning model known for its complexity. It consists of multiple layers of attention mechanisms and feedforward neural networks. The model's architecture involves intricate self-attention calculations, making it computationally intensive. Bi-LSTM is a recurrent neural network (RNN) variant that also contributes to the model's complexity. It includes forward and backward LSTM layers, which maintain hidden states and perform sequential computations. While not as complex as BERT, Bi-LSTM still adds computational load. One layer from Bi-LSTM is used in the proposed model.

For this research, the hardware used is Quadro RTX 5000 with GPU Memory 16 GB from the Paperspace website. The OLID dataset has 14,100 examples while the SOLID dataset has 1 million examples.

To evaluate the performance of the proposed model during the training, the early stopping technique is used. Early stopping is a technique used in machine learning to prevent overfitting and improve the generalization ability of a model. It involves monitoring the model's performance during training and stopping the training process when the performance on a validation set starts to degrade.

A validation set is typically used to measure the model's performance to implement the early stopping technique. The model is evaluated on the validation set after every training epoch or a certain number of iterations. If the performance on the validation set does not improve or starts to worsen consistently over a predefined number of epochs, the training process is stopped. During the training of the model, the weights are recorded after each epoch, and the loss function on the validation set is monitored to avoid overfitting. If the loss function on the validation set increases, the model will go back to the weights in the epoch that has the least loss function.

5 Discussion

This model is an extension of the model proposed by Al-Harigy et al. [11] for detecting cyberbullying using an augmented dataset. In the previous model we used SimCSE to learn new representations of the sentences and calculate the cosine similarity between labeled and unlabeled sentences to generate augmented sentences. The decision for assigning the labels to the unlabeled augmented sentences was taken from both SimCSE and the parallel BERT + Bi-LSTM model as SimCSE considers the semantic meaning only. Therefore, we needed to take the final decision from the parallel BERT + Bi-LSTM model to maintain understanding of the offensive meaning behind the sentence.

The focus of our work is to leverage CSSL for data augmentation to improve downstream tasks such as text classification by learning sentence representation in offensive language. This is more challenging because sentences that contain offensive language do not depend on the meaning of the word in isolation but rather on the meaning of the word in the context of the sentence, in addition to the use of emojis, which can take the meaning of the sentence in another direction. The purpose is to find a CSSL model which is capable of learning sentence representation based on the sentence context and utilize it for data augmentation to annotate datasets, which in turn could be used for the detection of cyberbullying. To maintain both the semantic and offensive meanings of the sentence when encoding the sentences and generating new representations, we retrained the sentence encoder (SimCSE) on new NLI datasets, which were created using labeled datasets.

As explained in Sect. 4, this research proposed a model for detecting cyberbullying by integrating the parallel BERT + Bi-LSTM model for detecting cyberbullying and CSSL represented by SimCSE for data augmentation. The proposed model consisted of different components which were discussed and explained in Sect. 4.

Several experiments were conducted in order to address the research questions as follows:

RQ1: How does the performance of the proposed Deep Contrastive Self-Supervised Learning (DCSSL) model based on pre-trained models compare with that of the baseline models for cyberbullying detection on social media?

RQ1 was addressed and validated by conducting the following experiments:

- The proposed parallel BERT + Bi-LSTM model, trained on SimCSE$_{\text{optimized}}$-augmented data, was compared with baseline models including the top three from SemEval-2020, BERT, and RoBERTa. It achieved the highest macro-averaged F1-score, outperforming all baselines. Incorporating the LP algorithm for assigning pseudo labels improved performance by nearly one percent over the top three SemEval-2020 models. Tables 6 and 7 show the comparison results.
- The proposed model was also trained on the Yelp dataset to validate its performance with other datasets, achieving high macro-averaged F1-scores using SimCSE$_{\text{optimized}}$ for data augmentation. Table 8 presents the performance results of the proposed model compared to baseline models BERT and RoBERTa on the Yelp dataset.

RQ2: How does the performance of the proposed DCSSL models using the augmented data compare with a manually labeled dataset for cyberbullying detection?

RQ2 was addressed and validated by conducting the following experiment:

The proposed model was trained on both OLID and OLID$_{\text{augmented}}$ to assess the impact of augmented data on cyberbullying detection. Results showed improved performance with the augmented data compared to the original training data. The results are presented in Table 5.

RQ3: How does the performance of CSSL for annotating dataset compare with the manually labelled dataset?

RQ3 was addressed and validated by conducting the following experiments:

- Experiments were conducted using SimCSE$_{\text{supervised}}$, SimCSE$_{\text{unsupervised}}$, and SimCSE$_{\text{optimized}}$ for data annotation, with LP for assigning pseudo labels. Results in Tables 2 show that using LP for pseudo labeling improves annotation performance.
- Experiments were conducted using SimCSE$_{\text{supervised}}$ and SimCSE$_{\text{optimized}}$ to validate these models for data annotation. The OLID training set and SOLID test set, both manually labeled, were used to compare pseudo labels generated by the models with true labels. SimCSE$_{\text{optimized}}$, fine-tuned on a new NLI dataset, improved performance by learning the semantic and offensive context of sentences. Results are presented in Table 3.
- Experiments were conducted to validate the performance of the proposed model parallel BERT + Bi-LSTM for cyberbullying detection using the fine-tuned SimCSE (SimCSE$_{\text{optimized}}$) for augmented data with the other versions of SimCSE. Results are presented in Table 4.
- From the literature, the CSSL approaches can be used for generating the augmented data either to generate new examples from the dataset as used by other researchers or to find the augmentations by calculating the similarities between two different datasets as used in this research. The CSSL approach is leveraged for data annotation by using a small labeled dataset as a seed and using the labeled examples to annotate unlabeled examples by finding similar examples which can improve the model's performance, e.g., in detecting cyberbullying, as discussed in this research. Furthermore, using CSSL for labeling the dataset incurred no labor cost and was not time consuming.
- Using CSSL alone for annotation is limited because it can learn sentence representations but cannot assign labels. Combining a small, labeled dataset with CSSL can efficiently annotate a large-scale dataset by leveraging both similarities and labels. CSSL-based models excel at learning from large amounts of unlabeled data, which is beneficial for tasks with limited labeled examples. This approach is particularly useful for data augmentation

in text classification and next sentence prediction. CSSL models need to be label-aware to be effective in downstream language tasks, which involve deeper abstractions and connotations.

6 Conclusion and future work

In this research, we proposed a model that utilized CSSL represented by SimCSE$_{\text{optimized}}$ for data augmentation between labeled and unlabeled offensive datasets. The SimCSE$_{\text{optimized}}$ model is a fine-tuned SimCSE which retrained on a new NLI dataset to force SimCSE to maintain the semantic and offensive meanings of the sentence while generating new representations for the sentences and calculating the similarity scores. The data augmented using SimCSE$_{\text{optimized}}$ was then used to train the model for cyberbullying detection.

The proposed model contains three components. The first component is creating a new NLI dataset using a labeled dataset (OLID) by calculating the similarity scores between every two sentences in OLID to find the positive pair sentence (entailment) and the hard negative (contradiction). The second component of the model is fine-tuning SimCSE (SimCSE$_{\text{optimized}}$) by retraining it on the new NLI dataset to force SimCSE to consider both the semantic and offensive meanings while generating the new representations for the sentences and calculating the similarities. SimCSE$_{\text{optimized}}$ is used to calculate the similarity between the sentences in the OLID and SOLID datasets to find the positive pair candidate for each sentence in OLID with the sentences in SOLID. The LP algorithm is used to assign the correct label for the augmented data, which are then added to OLID to create an augmented dataset OLID$_{\text{augmented}}$. The third component is the parallel BERT + Bi-LSTM model proposed in our previous work [11] which trained using the the augmented dataset OLID$_{\text{augmented}}$ for detecting cyberbullying. Both proposed models, proposed in the previous work [11] and the current work outperform the baseline models with F-1 scores of 0.9311 and 0.9231, respectively.

SimCSE$_{\text{optimized}}$ is validated with LP for data annotation, and the results are compared with the other versions of SimCSE: SimCSE$_{\text{supervised}}$ and SimCSE$_{\text{unsupervised}}$. The results show that SimCSE$_{\text{optimized}}$ with LP outperforms the other versions of SimCSE with LP. The performance of the proposed model in detecting cyberbullying using the augmented dataset has a macro average F1-score of 0.9231, outperforming the baseline models.

Future work may involve utilizing large versions of the transformers, such as BERT$_{\text{large}}$, and comparing the effect of increasing the transformers' layers and parameters on the performance of cyberbullying detection with the proposed model using BERT$_{\text{base}}$.

Author contributions All authors reviewed the manuscript

Data availability The datasets analyzed during the current research are competition-related datasets. The OLID dataset is the official dataset of the SemEval-2019 Task 6: Identifying and Categorizing Offensive Language in Social Media (OffensEval, 2019) [33]. The SOLID dataset is the official dataset of the SemEval-2020 Task 12 shared task: Multilingual Offensive Language Identification in Social Media (OffensEval, 2020) [34]– The competition [34] distributed only the tweets IDs for the SOLID training set. We used ~ 1 million tweets of the SOLID training set crawled by [11]. All the datasets used in this research including the generated dataset are uploaded to the repository and are available upon request.

Declarations

Conflict of interest The authors declare no conflict of interest.

References

1. Taylor, P.: "Statista. [Online]. Available: https://www.statista.com/statistics/1190263/internet-users-worldwide/. Accessed 5 February 2023
2. Kumar, A., Sachdeva, N.: Multimodal cyberbullying detection using capsule network with dynamic routing and deep convolutional neural network. Multimed. Syst. **8**(6), 2043–2052 (2022)
3. Kumar, A., Sachdeva, N.: A Bi-GRU with attention and CapsNet hybrid model for cyberbullying detection on social media. World Wide Web **25**, 1537–1550 (2021)
4. Al-Harigy, L.M., Al-Nuaim, H.A., Moradpoor, N., Tan, Z.: "Building towards automated cyberbullying detection: a comparative analysis,." Comput. Intell. Neurosci. **2022**, 4794227 (2022)
5. Wang, X., Qi, G.-J.: Contrastive learning with stronger augmentations. IEEE Trans. Pattern Anal. Mach. Intell. **45**(5), 5549–5560 (2022)
6. Miyai, A., Yu, Q., Ikami, D., Irie, G., Aizawa, K.: "Rethinking rotation in self-supervised contrastive learning: adaptive positive or negative data augmentation," in IEEE/CVF Winter Conference on Applications of Computer Vision, (2023)
7. Falcon, W., Cho, K.: "A framework for contrastive self-supervised learning and designing a new approach," in arXiv preprint arXiv, (2020)
8. Saunshi, N., Ash, J. T., Goel, S., Misra, D., Zhang, C., Arora, S., Kakade, S., Krishnamurthy, A.: "Understanding contrastive learning requires incorporating inductive biases," in the 39 the International Conference on Machine Learning, (2022)

9. Xiao, T., Wang, X., Efros, A. A., Darrell, T.: "What should not be contrastive in contrastive learning," in The 9th International Conference on Learning Representations, (2021)
10. Fang, H., Wang, S., Zhou, M., Ding, J., Xie, P.: "CERT: Contrastive self-supervised learning for language understanding," in arXiv preprint arXiv:2005.12766, (2020)
11. Al-Harigy, L., Al-Nuaim, H., Moradpoor, N.: "Deep pre-trained contrastive self-supervised learning: a cyberbullying detection approach with augmented datasets," in 14th International Conference on Computational Intelligence and Communication Networks (CICN), Al-Khobar - KSA, (2022)
12. Gao, T., Yao, X., Chen, D.: "SimCSE: simple contrastive learning of sentence embeddings," In: Conference on Empirical Methods in Natural Language Processing, (2021)
13. Zampieri, M., Malmasi, S., Nakov, P., Rosenthal, S., Farra, N., Kumar, R.: "Predicting the type and target of offensive posts in social media," in Proceedings of the 2019 conference of the north american chapter of the association for computational linguistics: human language technologies, (2019)
14. Rosentha, S., Atanasova, P., Karadzhov, G., Zampieri, M., Nakov, P.: "SOLID: a large-scale semi-supervised dataset for offensive language identification," in arXiv:2004.14454, (2021)
15. Chen, T., Kornblith, S., Norouzi, M., Hinton, G.: "A simple framework for contrastive learning of visual representations," in The 37th International Conference on Machine Learning (ICML'20), (2020)
16. Wu, Z., Wang, S., Gu, J., Khabsa, M., Sun, F., Ma, H.: "CLEAR: contrastive learning for sentence representation," in arXiv preprint arXiv:2012.15466, (2020)
17. Giorgi, J., Nitski, O., Wang, B., Bader, G.: "DeCLUTR: deep contrastive learning for unsupervised textual representations," in "Italic">arXiv preprint arXiv:2006.03659, (2020)
18. Chen, Q., Zhang, R., Zheng, Y., Mao, Y.: "Dual contrastive learning: text classification via label-aware data augmentation," in arXiv preprint arXiv:2201.08702, (2022)
19. Mao, Z., Zhu, D., Lu, J., Zhao, R., Tan, F.: "SDA: simple discrete augmentation for contrastive sentence representation learning," in arXiv preprint arXiv:2210.03963, (2022)
20. Chen, J., Zhang, R., Mao, Y., Xu, J.: "ContrastNet: a contrastive learning framework for few-shot text classification," in The Thirty-Sixth AAAI Conference on Artificial Intelligence (AAAI-22), (2022)
21. Febriana, T., Budiarto, A.: "Twitter dataset for hate speech and cyberbullying detection in indonesian language," in International Conference on Information Management and Technology (ICIMTech), (2019)
22. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: "BERT: pre-training of deep bidirectional transformers for language understanding," arXiv preprint arXiv:1810.04805, (2018)
23. Radford, A., Narasimhan, K., Salimans, T., Sutskever, I.: Improving Language Understanding by Generative Pre-Training. OpenAI (2018)
24. Lewis, M., Liu, Y., Goyal, N., Ghazvininejad, M., Mohamed, A., Levy, O., Stoyanov, V., Zettlemoyer, L.: "BART: denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension," in arXiv preprint arXiv:1910.13461, (2019)
25. Paul, S., Saha, S.: CyberBERT: BERT for cyberbullying identifcation. Multimed. Syst. **28**(6), 1897–1904 (2020)
26. Elsafoury, F., Katsigiannis, S., Pervez, Z., Ramzan, N.: When the timeline meets the pipeline: a survey on automated cyberbullying detection. IEEE access **9**, 103541–103563 (2021)
27. Guo, X., Anjum, U., Zhan, J.: "Cyberbully detection using BERT with augmented texts," in International Conference on Big Data (Big Data), (2022)
28. Tripathy, J.K., Chakkaravarthy, S.S., Satapathy, S.C., Sahoo, M., Vaidehi, V.: ALBERT-based fine-tuning model for cyberbullying analysis. Multimed. Syst. **28**, 1941–1949 (2020)
29. Nouri, N.: "Data augmentation with dual training for offensive span detection," in the 2022 conference of the north american chapter of the association for computational linguistics: human language technologies, (2022)
30. Gonzalez-Pizarro, F., Zannettou, S.: "Understanding and detecting hateful content using contrastive learning," in arXiv preprint arXiv:2201.08387, (2022)
31. B. Bhatia, A. Verma, Anjum and R. Katarya, "Analysing Cyberbullying using Natural Language Processing by Understanding Jargon in Social Media," *Sustainable Advanced Computing*, Springer, Singapore (2022)
32. Balakrishnan, V., Khan, S., Arabnia, H.R.: Improving cyberbullying detection using twitter users' psychological features and machine learning. Comput. Secur. **90**, 101710 (2020)
33. Zampieri, M., Malmasi, S., Nakov, P., Rosenthal, S., Farra, N., Kumar, R.: "SemEval-2019 Task 6: identifying and categorizing offensive language in social media (OffensEval)," The 13th International Workshop on Semantic Evaluation, (2019)
34. Zampieri, M., Nakov, P., Rosenthal, S., Atanasova, P., Karadzhov, G., Mubarak, H., Derczynski, L., Pitenis, Z., Coltekin, C.: "SemEval-2020 task 12: multilingual offensive language identification in social media (OffensEval 2020)," in The Fourteenth Workshop on Semantic Evaluation, (2020)
35. Li, B., Hou, Y., Che, W.: "Data augmentation approaches in natural language processing: a survey," in Ai Open 3, (2022)
36. Wiedemann, G., Yimam, S. M., Biemann, C.: "UHH-LT at SemEval-2020 task 12: fine-tuning of pre-trained transformer networks for offensive language detection," in The International Workshop on Semantic Evaluation (SemEval), (2020)
37. Wang, S., Liu, J., Ouyang, X., Sun, Y.: "Galileo at SemEval-2020 task 12: multi-lingual learning for offensive language identification using pre-trained language models," in the International Workshop on Semantic Evaluation (SemEval)., (2020)
38. Dadu, T., Pant, K.: "Team rouges at SemEval-2020 task 12: cross-lingual inductive transfer to detect offensive language," in the International Workshop on Semantic Evaluation (SemEval), (2020)
39. Zhang, X., Zhao, J., LeCun, Y.: "Character-level convolutional network for text classification applied to chinese corpus," in "Italic">arXiv preprint arXiv:1611.04358, (2016)

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.