

UNDERGRADUATE PROJECT REPORT

|  |  |
| --- | --- |
| **Project Title:** | **Toxic Comments Classification Based on Deep Learning** |
| **Surname:** | **Gao** |
| **First Name:** | **Ruiling** |
| **Student Number:** | **201918020301** |
| **Supervisor Name:** | **Joojo Walker** |
| **Module Code:** | **CHC 6096** |
| **Module Name:** | **Project** |
| **Date Submitted:** | **May 5, 2023** |

# Declaration

# Acknowledgment

I want to express my gratitude to my thesis advisor, Joojo Walker, for his unwavering support and academic guidance throughout the topic selection and writing process. Whenever I was anxious about whether my arguments needed further refinement, he always reminded me to focus on myself and not be influenced by external factors.

I am also grateful to my friends. I met my friends in my first-year classes, and every moment spent with them on aimless walks in the wee hours of the morning was the best antidote for my anxiety. I would also like to thank my friends who I have known for nearly ten years. I remember before we entered university, we made a promise to meet at least a few times every year, even if we went to different schools. Four years have passed, and now we can still find a barbecue stall and chat over a beer. I also want to express my gratitude to some friends who have left a trace in my life and appreciate every moment we spent together.

Most importantly, I am grateful to my family, from whom I gained the confidence of being loved, especially my mother. She is not a person who is good at expressing herself and always says awkward things with a kind heart, but I know that underneath it all is a deep and warm love. I appreciate her meticulous care and support for me to pursue whatever I want to do, and I love her with all my heart.

Lastly, I would like to thank myself for not giving up and working hard and playing hard for the past four years. I will remember myself sitting at the corner table in the reading room all day, pulling all-nighters to finish coursework. I will remember being anxious during application season and unable to sleep every night, feeling complex emotions after being rejected, and feeling ecstatic after receiving my ideal offer, and I will also remember the self who made the final decision. These experiences have shaped who I am at 22, and I am grateful for them all.

# Table of Contents

[Declaration i](#_Toc133322902)

[Acknowledgment ii](#_Toc133322903)

[Table of Contents iii](#_Toc133322904)

[Abstract v](#_Toc133322905)

[Abbreviations vi](#_Toc133322906)

[Glossary vii](#_Toc133322907)

[Chapter 1 Introduction 1](#_Toc133322908)

[1.1 Background 1](#_Toc133322909)

[1.2 Aim 2](#_Toc133322910)

[1.3 Objectives 2](#_Toc133322911)

[1.4 Project Overview 2](#_Toc133322912)

[1.4.1 Scope 2](#_Toc133322913)

[1.4.2 Audience 3](#_Toc133322914)

[Chapter 2 Background Review 4](#_Toc133322915)

[Chapter 3 Methodology 6](#_Toc133322916)

[3.1 Approach 6](#_Toc133322917)

[3.1.1 Prepare the Input Data 6](#_Toc133322918)

[3.1.2 Pre-process the Data 6](#_Toc133322919)

[3.1.3 Build the Deep Learning Model 9](#_Toc133322920)

[3.1.4 Testing 11](#_Toc133322921)

[3.2 Technology 12](#_Toc133322922)

[3.3 Project Version Management 12](#_Toc133322923)

[Chapter 4 Results 14](#_Toc133322924)

[4.1 Experimental Settings 14](#_Toc133322925)

[4.1.1 Dataset Statistics Details 14](#_Toc133322926)

[4.1.2 Evaluation Metrics 16](#_Toc133322927)

[4.1.3 Baseline Models 17](#_Toc133322928)

[4.1.4 Parameter Settings 18](#_Toc133322929)

[4.2 Performance Comparison 18](#_Toc133322930)

[Chapter 5 Professional Issues 27](#_Toc133322931)

[5.1 Project Management 27](#_Toc133322932)

[5.1.1 Activities 27](#_Toc133322933)

[5.1.2 Schedule 29](#_Toc133322934)

[5.1.3 Project Data Management 30](#_Toc133322935)

[5.1.4 Project Deliverables 30](#_Toc133322936)

[5.2 Risk Analysis 30](#_Toc133322937)

[5.3 Professional Issues 33](#_Toc133322938)

[5.3.1 Legal Issue 33](#_Toc133322939)

[5.3.2 Ethical Issue 33](#_Toc133322940)

[5.3.3 Social Issue 33](#_Toc133322941)

[5.3.4 Environment Issue 34](#_Toc133322942)

[Chapter 6 Conclusion 35](#_Toc133322943)

[6.1 Conclusion and Reflection 35](#_Toc133322944)

[6.2 Future Work 35](#_Toc133322945)

[References 36](#_Toc133322946)

# Abstract

With the growth of online communities on the Internet, a large amount of cluttered textual information floods the online environment, some users post anti-social and malicious comments on online platforms in an attempt to undermine the usability of the Internet. Therefore, text classification is necessary for detecting inappropriate comments, and methods such as deep learning models, particularly neural networks, are becoming increasingly popular and effective for this task. Long Short-Term Memory neural network models have addressed the limitations of traditional recurrent neural network models by incorporating long-term memory. As a result, LSTM models have been widely utilized in various text classification tasks and have shown significant performance improvements. The main goal of this project is to design implementations of different deep learning-based models for the detection and classification of toxic comments. Furthermore, compare models to get a better result. Long Short-Term Memory model and Gated Recurrent Unit model were implemented to solve the problem. The experimental findings demonstrate that the model put out in the research has the capability of comprehending semantic data and identifying probable textual malice. In this project, the accuracy rate for the Gated Recurrent Unit model is 97.83%, while the accuracy rate for Long Short-Term Memory model is 97.72% for identifying the objectionable comment dataset.

***Keywords: Text Classification, Deep Learning Model, Long-Short Term Memory, Gated Recurrent Unit***

# Abbreviations

| Abbreviations | Definition |
| --- | --- |
| TC | Text classification |
| SA | Sentiment Analysis |
| LSTM | Long Short-Term Memory |
| RNN | Recurrent Neural Network |
| NLTK | Natural Language Toolkit |
| CEC | Constant Error Carousels |
| GRU | Gated Recurrent Unit |
| CPU | Central Processing Unit |
| GPU | Graphic Processing Unit |
| P | Precision |
| R | Recall |
| TP | True Positive |
| FN | False Negative |
| FP | False Positive |
| TN | True Negative |
| NB-SVM | Support Vector Machine builds on Naive Bayes |
| ACC | Acceptable |
| RFI | Require Further Investigation |
| N/ACC | Not Acceptable |
| GDPR | General Data Protection Regulation |

# Glossary

**Text Classification**: a task that involves automatically assigning predefined categories or labels to textual documents, based on their content. The goal of text classification is to enable automated organization, filtering, and analysis of large collections of textual data, such as social media posts, news articles, customer feedback, and legal documents.

**Deep Learning Model**: a type of artificial neural network that is designed to learn and extract features from complex and large-scale data sets. It is typically composed of multiple layers of interconnected nodes, each layer learning to identify increasingly abstract features in the data. Deep learning models are trained using large amounts of data and can be used for a variety of tasks such as classification, regression, and generation of new data.

**LSTM**: it is a type of recurrent neural network architecture that is designed to overcome the vanishing gradient problem that can occur in traditional RNNs.

**GRU**: GRU is a variant of a recurrent neural network that consists of two gating units, an update gate, and a reset gate. the update gate of GRU determines how much information should be forgotten from the previous hidden state at the current time step and how much new information needs to be updated.

**Bi-LSTM**: Bi-LSTM captures context more comprehensively by processing both forward and reverses sequence information. It consists of two mutually independent LSTM layers, with one LSTM processing the input sequence from left to right and the other LSTM processing the input sequence from right to left. The outputs of the two LSTM layers are joined to form the final Bi-LSTM model output.

**Attention Mechanism**: Attention mechanism refers to a computational model in which the system learns to focus its attention on certain parts of input data, typically a sequence of vectors or a matrix while performing a task. The attention mechanism computes a weight for each input element, indicating its importance or relevance to the task being performed. This allows the model to selectively process certain parts of the input while ignoring others.

# Introduction

## Background

With the advent of the digital age, the number of electronic documents is increasing. One of the more typical phenomena is that with the growth of online communities on the Internet, a large amount of cluttered textual information, such as posts, comments, etc., floods the entire online environment. Most normal and virtuous netizens are spontaneously maintaining the sustainability and usability of the Internet, but in contrast, some users post anti-social and malicious comments on online platforms in an attempt to undermine the usability of the Internet [1]. The emergence of malicious comments violates the legitimate rights of netizens and can cause serious mental and psychological harm to them. Therefore, the efficient management and processing of large volumes of comment texts have become a target of interest for researchers [2].

Text classification is one of the effective ways to locate and triage information efficiently and accurately, solving the problem of information clutter as much as possible [3]. Text classification (TC), also known as text categorization, is an extensive area of current research in linguistic text mining and processing. TC is a process that uses deep learning algorithms to categorize text content into pre-given sets of labels [4]. Deep learning-based text classification techniques have been developed and matured since the 1990s. Compared to text classification systems based on knowledge engineering and expert systems, classification techniques using deep learning provide better classification results and flexibility and have become the main techniques used in related fields [5]. Among the techniques for text classification by different criteria, sentiment analysis (SA), also known as opinion mining, is a branch of text classification. Its main function is to identify and analyze the sentiment in a text by using pre-given labels with human sentiment colors and sentiment tendencies, such as positive, negative, neutral, etc [6].

The report is divided into six sections and the structure of the report with the main content of each section is organized as follows. The first section introduces the basic concepts of text classification, the purpose and significance of this research, an analysis of the problems addressed by text classification, and an overview of the research on the topic. Finally, the overall structure of the report is given. Section 2 introduces the research background of text classification and summarizes the current state of research and the main features of text classification. Section 3 presents the main research methods chosen for the topic, the techniques used, the data, and the model testing strategy. Section 4 gives the results of each experiment, while Section 5 shows the project management plans including the activities, schedules, data, and version management plans with the potential risks which may appear during the process of the research and the relevant legal, social, ethical, and environmental issues in the context of the project. The final section presents the achievements accomplished and future works of the project.

## Aim

The main goal of this project is to design implementations of different deep learning-based models for the detection and classification of toxic comments. Furthermore, compare models to get a better result.

## Objectives

The objectives of this text classification project are as follows:

1. Conduct background research on text classification, understand the field of Natural Language Processing and the corresponding deep learning technologies used for sentiment analysis.
2. Collect usable toxic comments dataset from the Internet.
3. Clean and pre-process the data for modeling.
4. Extract features from the text in the cleaned datasets.
5. Implement different models using datasets and assess the quality of the models.
6. Analyze the quality of the models and compare the strengths and weaknesses of each model.
7. Develop data and model testing and evaluation strategy.
8. Analyze risks based on the current project.

## Project Overview

### Scope

The project is designed to analyze the sentiment of comments made by users on the talk page of an online encyclopedia website named Wikipedia, filtering out malicious comments and classifying them into different categories, such as hate speech, personal attacks, pornography, violence, etc. The project helps social network staff to automatically screen out, therefore, manage malicious comments, reducing labor and time costs, while also helping to clean up the online environment.

### Audience

Text classification is one of the effective management to help to locate and triage information efficiently and accurately, solving the problem of information clutter as much as possible[3].

# Background Review

To date, several models have been used in research on sentiment classification. Among them, Kong F and Chen G have presented a neural network model combining Word2Vec [7] with Bi-LSTM[8] to learn the spatial representation of word vectors through Word2Vec, transforming the text into a sentence representation in the input layer feature space, and improve the network using constant mapping covariance theory. The model using the improved Bi-LSTM is able to present excellent improvements in the dataset.

Ding Y has proposed a classification model based on an attentional mechanism called ON-LSTM [9]. The method is mainly based on a transfer learning approach through feature extraction, where the performance of the model is tuned to the best in the source dataset and then applied to the test set. A multi-level embedding model under the attentional selection mechanism is also proposed. Through the embedding representation at the character level and sentence level in addition to the word level, information that is more conducive to classification in the text can be extracted[10].

Li G and Guo J have proposed a new architecture of bidirectional LSTM (Bi-LSTM) [8] with Attention Mechanism and convolutional layer, which can more precisely extract text semantics and achieve better text classification results. In AC-BiLSTM model, the convolutional layer is used to retrieve higher-level phrase representations from word embedding vectors, and Bi-LSTM layer is used to gain access to the forward and backward contexts. The attention mechanism is applied to put more focus on the important information in the output of the hidden layer. A SoftMax classifier is ultimately used to categorize the text information after processing. The strength of the AC-BiLSTM model lies in its ability to both extract partial features of phrases and to understand the semantics of phrases within a sentence [11].

Khataei Maragheh M et al. proposed a new model called SHO-LSTM based on LSTM and Spotted Hyena Optimize (SHO) algorithm[12]. Firstly, the Skip-gram method is used for the embedding of word vectors, and subsequently, the SHO algorithm is used to optimize the initial weights of the LSTM. In the SHO algorithm, every response to the problem is encoded as a hyena in this procedure. Following a leading hyena, the hyenas are then guided to the best response. During the exploring process, it is discovered that SHO has the advantages of a quick rate of convergence and an excellent balance between both mining and exploration.

Onan A and Tocoglu M proposed a stacked three-layer bidirectional long short-term memory architecture to identify sarcastic text documents[13]. They start by using the inverse gravity moment-based term weighted word embedding model with n-grams to create a weighted word embedding scheme for representing text documents. This scheme is then fed into a stacked bi-directional LSTM architecture, which extracts contextual information and processes the text documents. The outputs of the final hidden layer are concatenated and passed through a SoftMax layer to determine the class label (sarcasm or non-sarcasm). The model shows promising results for the sarcasm identification task, achieving a classification accuracy of 95.30%.

A model for analyzing the sentiment of user comments called Co- LSTM, is proposed by Behera R et al.[14], which combines a convolutional layer and an LSTM layer, enabling to some extent the advantages of local feature extraction with the convolutional layer, while also using the LSTM Layer to enable the model to make contextual connections to the text. The input feature vector is embedded using a pre-trained word embedding model, avoiding the need for domain-specific specialized vocabulary. A CNN layer is used before the LSTM layer to identify important features from the embedding vector, significantly reducing training time. Finally, the use of LSTM layers allows the model to examine the sequential arrangement of comments, enabling better analysis of sentiment.

For a summarized version of the background review of the features of existing approaches of implementation using recurrent neural network for text classification, refer to Table 1.

| **Model** | **Recall Ratio** | **Precision Ratio** | **F1-Score** | **Data processing** |
| --- | --- | --- | --- | --- |
| Bi-LSTM + Word2Vec | 97.00% | 89.00% | 92.00% | Word2Vec |
| Bi-LSTM | / | / | 92.79% | Attention selection mechanism + Fine-grained text classification |
| AC-BiLSTM | 87.81% | 87.28% | 86.45% | Attention Mechanism + Convolutional Layer |
| SHO-LSTM | 89.74% | 89.74% | 87.96% | Skip-gram + Spotted Hyena Optimizer |
| 3 Layer Bi-LSTM | / | / | / | N-grams + Stacked bi-directional LSTM |
| CO-LSTM | 83.02% | 83.50% | 83.13% | Convolutional Layer + Word Embedding |

Table 1 Existing Approaches and Features

# Methodology

## Approach

This report is designed to investigate the effectiveness of different deep learning models for predicting malicious comments by modeling the same dataset and comparing the accuracy of different models. This section describes the entire flow of the experiment shown in Figure 1, as well as the basic principle and algorithm of the core, RNN-based deep learning models used in this paper to analyze user-posted comments on the Wikipedia talk page. The two models are used to determine whether a comment contains malicious words and to classify comments that contain malicious words into different categories.

图示

描述已自动生成

Figure 1 Flow Diagram

### Prepare the Input Data

Wikipedia, a multilingual encyclopedic collaborative project based on wiki technology, whose headquarter is based in the United States, is an online encyclopedia written in several languages[15]. There is an online forum page called the talk page (discussion page) in Wikipedia where editors can post comments about articles or other Wikipedia pages in talk pages (also known as discussion pages).

Kaggle, a platform for developers and data scientists to run machine learning competitions, host databases, and write and share code, has crawled comments of approximately 130M in size from the Wikipedia talk page and has collated these comments into a dataset in CSV format for different kinds of model training[16]. In this paper, a set of classified comments of 59.8M was used as the input training set and a set of unclassified comments of approximately 70M was used as the test set to evaluate the performance of different deep learning models.

### Pre-process the Data

The entire dataset contains the ids of users who posted the comment, the content of the comments, and different types of labels of the malicious comments in the form of one-hot codes. For the contents of the comments analyzed as textual data in this project, the dataset may contain duplicate or irrelevant observations, structural errors, or unwanted outliers. Consequently, some pre-processing of the raw data is applied.

All letters were firstly converted into lowercase for better feature extraction. Then, in order to visualize the cleaned data, the number of words per comment is displayed as a frequency chart. Next, useless words in the content that do not have an impact on the overall experiment are removed to improve the quality of the text. These words, which usually have no real meaning, such as pronouns, prepositions, or conjunctions, are collectively referred to as stop words[17]. Natural Language Toolkit (NLTK), the most used word segmentation package developed at the University of Pennsylvania, was called to remove these common stop words from the comments. The word frequency chart and wordcloud of all the malicious and non-malicious comments of the entire training set are displayed separately below.

图形用户界面, 应用程序, Teams

描述已自动生成

Figure 2 Word Frequency

文本

描述已自动生成

Figure 3 WordCloud for Toxic Comments

文本

描述已自动生成

Figure 4 WordCloud for non-Toxic Comments

After the overview of the data and initial processing, most of the comments in the entire dataset are concentrated in less than 500 characters, so data pre-processing operations are applied to retain the important information in the comments for better word separation operations: The comment text was converted into processed text input by limiting its length and size. From the below histogram showing the word length after the pre-processing progress, it is evident that most of the sentences fall in the range of 1-200 lengths, so the max length of 200 words was kept when padding the sentence. As the last step of preprocessing, each comment was vectorized from text into an array of integers with the help of a tokenizer, and any comment above the max length was trimmed, and zeros were padded in all the sentences below 200.

图形用户界面, 应用程序

描述已自动生成

Figure 5 Word Length

### Build the Deep Learning Model

In this project, recurrent neural networks (RNN) are used as the main model. The use of RNN networks dated to the 1980s and has evolved into one of the main network models in the field of deep learning[18]. One thing that distinguishes RNN from other neural network is that RNN can be applied to scenarios where the input of this network can be a sequence of sequence type of data. In other words, the output of an RNN network is somewhat related to its previous output. Due to its recurrent feedback connections, the RNN model can establish relationships between the inputs of the preceding and following sequences, allowing the output values of the RNN at each moment to be influenced by the input values of multiple previous moments, yielding more accurate results[19]. Different RNN models were used in this project to compare their performance. The structure of the RNN is illustrated in Figure 6.

图表, 散点图

描述已自动生成

Figure 6 Structure of RNN

Long Short-Term Memory (LSTM) [20] is one of the RNN variants created to solve the RNN gradient vanishing problem. LSTM has a linear unit called constant error carousels (CECs) and is controlled by three gates that are used to store the input into the model from real-time information entered into the model[21]. The input gate controls whether information from the current moment is allowed to be added to the CEC, the output gate controls whether information from the previous moment's CEC will be output to affect the output of the next moment's node, and the forget gate controls whether information from the current moment's CEC will be formatted. Figure 6 illustrates an architecture of a single LSTM unit, where c represents the entire memory cell and a represents the output of this single unit.

图示

描述已自动生成

Figure 7 Architecture of LSTM Unit

The calculation of an individual cell of the LSTM will be shown as Eq. (1) ~ Eq. (6):

Different weights wi, wf, w0, ui, uf, u0 and biases bi, bf, and b0 in each gate are computed with the input values xt at time t and the cell state vector vt-1 at time t-1 to form the input values zi, zf and z0 for each gate. These input values are computed into values between 0 and 1 utilizing a specific activation function f(zi), f(zf) and f(z0) for mimicking the extent of openness of the gate.

In an identical way, the input of the LSTM cell z enters the cell through a specific activation function g(z).

As the value is stored in the memory cell, the update of c is displayed in Eq. (5).

The updated c is then calculated with the activation function h(c), and multiplied with f(z0) to yield the final output a of the LSTM unit at time t.

Another model used in this report is called Gated Recurrent Unit neural network (GRU). GRU [22] is a neural network that improves on the LSTM by not only retaining the gate feature but also simplifying it by reducing the three gates in the model to two gates: the update gate and the reset gate[19]. The update gate is used to control how much of the information in the CEC from the previous moment will be output and thus affect the output of the node at the next moment, while the reset gate is used to control how much information can be added to the CEC. Figure 8 shows a schematic diagram of the GRU structure.

图示

描述已自动生成

Figure 8 Schematic Diagram of GRU

The calculation of an individual cell of the GRU will be shown as Eq. (7) ~ Eq. (10):

Value s(uu) and s(ur) of Update Gate and Reset Gate are formed by having input value xt at time t and the cell state ht-1 at moment t-1 firstly calculated with weights wu and wr and the bias bu and br of teach gate, and then calculated with the corresponding activation function s.

Then subsequent judgment is made using the update gate to determine the degree to which the new input value can be stored in the candidate where ⊙ indicates an all-element multiplication.

The reset gate is used to do the determination to choose whether to store the candidate as the new value in the memory cell.

### Testing

Two generic model-like tests, pre-train and post-train tests, are written[22].

#### Pre-train Test

Some tests can be used to test the data without adjusting the parameters.

* Check if the type of labels of the training and testing sets are the same one-hot code
* Check if the data of the training and testing sets are both containing id and comment\_text
* Check if the output of the LSTM and GRU model matches all types in the label
* Check that the range of the LSTM and GRU model output matches the range of the label of 0 to 1
* Adding assertions to the model to control the operation of the model

#### Post-train Test

* Invariance Tests: Manually make changes (e.g.: change “a” to “A”) to the data entered in the LSTM and GRU model to see if there is an impact on the model's predictions while ensuring that the output is not affected
* Directional Expectation Test: Manually make changes (e.g.: change “hate” to “like”) to the data entered in the LSTM and GRU model to see if there is an impact on the model's predictions if it interferes with the model output
* Data Unit Test: Classify possible erroneous results in the model

Once these processes had been done, the evaluation and testing of the model can be used as a basis for modifying and refining the model.

## Technology

The experimental environment used in this paper: M1 ProM1 Pro integrates several different components, including CPU, GPU, Neural Engine, etc.

The 8-core M1 Pro is equipped with a 14-core GPU and a 16-core neural engine for machine learning.

Python 3.8, and TensorFlow 2.7.0 with Jupyter Notebook are used to implement the methods.

| **Hardware** | **Software** |
| --- | --- |
| CPU: M1 Pro(8-core) | Python 3.8 |
| GPU: Mi Pro(14-core) | TensorFlow 2.7.0 |
| Neural Engine: M1 Pro | Jupyter Notebook |

Table 2 Experimental Environment

## Project Version Management

Versions of the project are stored in GitHub:

Progress of the project can be seen in the sharing folder with the URL <https://github.com/Ivvvvvvvy/OBU_Project>

# Results

## Experimental Settings

### Dataset Statistics Details

The experimental dataset used in this paper was adopted from a Kaggle competition called the "toxic comment classification challenge" [16]. The entire dataset was divided into two parts, a test set and a training set, containing 159571 and 153164 comments posted on the Wikipedia talk page and the ids of the posters, respectively. The training set is divided into two parts: the validation set and the training set. A preview of the training and test sets is shown in the figure below.

表格

中度可信度描述已自动生成

Figure 10 Samples of the training set

表格

中度可信度描述已自动生成

Figure 11 Samples of the test set

In the training set, comments ruled as malicious or other violations were labeled as 1 in the category in which they were judged, and normal comments were labeled as 0. Comments judged to be malicious accounted for roughly 10% of the overall dataset. The content structure of the training dataset and test set are shown in the figure below.

| **Type** | **train set** | **validation set** | **Total** |
| --- | --- | --- | --- |
| normal | 129049 | 14297 | 143346 |
| toxicity | 5078 | 588 | 5666 |
| severe toxic | 1446 | 149 | 1595 |
| obscene | 7587 | 862 | 8449 |
| threat | 442 | 36 | 478 |
| insult | 7065 | 812 | 7877 |
| identity hate | 1268 | 137 | 1405 |

Table 3 Training Set Structure

| **Type** | **Test Set** |
| --- | --- |
| normal | 143346 |
| toxicity | 6090 |
| severe toxic | 367 |
| obscene | 3691 |
| threat | 211 |
| insult | 3427 |
| identity hate | 712 |

Table 4 Test Set Structure

A typical example of each type of malicious comments in the dataset is shown in the figure below. As can be seen from the example, the categories of violation include severe toxic, obscene, threat, insult and identity hate. In the dataset, these different types of offending comments are labeled 1 for the specific category to which they belong and 0 for the rest of the labels.

| **Type** | severe toxic | obscene | threat | insult | identity hate |
| --- | --- | --- | --- | --- | --- |
| **Samples of toxic data** | Die you piece of ordure You do not deserve to be on this planet. 137.205.183.70 | “Blocked by Freestyle frappe. Yeah, that's right, blocked for saying "IMHO WMC is a jerk." Personal insult, huh? Sock puppet? I'm fucking steamed, Freestyle frappe. And coming right after WMC's admin confirmation? If you're trying to punish me this is some of the most ridiculous bullshit gamesmanship, I've ever seen | I’ll kill you all!!!! | I'm just getting recent pics of somebody & an uneducated dufus came & removed my recent pics so help me get recent pics & try reporting that dufus & that idiot's user is Lil crazy thing. User:Pic Business | The conclusion is obvious - when you don't like what someone says because it's true, do what the Nazis did and silence that person. How Jewish of you! 12.176.152.194 |

Table 5 Samples of Malicious Comments

### Evaluation Metrics

This subsection discusses the performance evaluation of the proposed model. The metrics used to evaluate the proposed model are accuracy and loss. From this subsection, the evaluation of how well the model will perform will be demonstrated.

Certain criteria are used to evaluate the performance of each model in this experiment, namely the Precision(P), the Recall (R), and the harmonic mean of Precision, and Recall (F1-score). After calculating these criteria for each category, macro-F1, and Accuracy are used to evaluate the performance of the whole model.

In particular, P represents the number of samples with positive predictions that are predicted correctly, and R refers to how many of the positive samples in the dataset are predicted correctly. The confusion matrix representing the actual and predicted values is shown in the table below.

|  |  |  |
| --- | --- | --- |
| Actual  Predict | Positive | Negative |
| Positive | True Positive (TP) | False Negative (FN) |
| Negative | False Positive (FP) | True Negative (TN) |

Table 6 Confusion Matrix

In the table, TP: samples that are positive and predicted to be positive; FN: samples that are positive but predicted to be negative; FP: samples that are negative but predicted to be positive; TN: samples that are negative and predicted to be negative.

Therefore, the expressions of the evaluation criterion of the model are as (11) to (15):

### Baseline Models

The following baseline methods for text classification are benchmarked in this study. They are useful methods that have produced positive outcomes in text classification:

GloVe + GRU: A Gated Recurrent Unit model using GloVe to perform word vectorization[23].

Transformers: a model structure that completely hinges on an attention mechanism, forgoing recurrence, to identify global dependencies between inputs and outcomes[24].

LSTM + NB-SVM: A Long Short-Term Memory model which combines Support Vector Machine builds on naive Bayes features[25].

Naïve Bayes + Logistic Regression: Logistic Regression model building on naive Bayes features.

Random Forest: An algorithm integrates multiple decision trees into a forest, where each decision tree is a classifier, and the random forest will aggregate all the classification votes to predict the outcome[26].

LSTM: Long Short-Term Memory.

GRU: Gated Recurrent Unit.

| **Model** | **Accuracy** | **Reported In** |
| --- | --- | --- |
| GloVe + GRU | 97.97% | Pratap D[27] |
| Transformers | 98.34% | Afzal B[28] |
| LSTM + NB-SVM | 98.10% | Howard J[29] |
| Naive Bayes + Logistic Regression | 97.95% | Abhi[30] |
| Random Forest | 96.62% | Zhu D[29] |
| LSTM | 99.16% | Project Result |
| GRU | 97.73% | Project Result |

Table 7 Baseline Model

### Parameter Settings

The parameters used in this paper are shown in the table below. The dimension of the word vector obtained after the embedding layer is 13, the epochs are set to 2, the hidden unit size is 50 for both LSTM and GRU, and the batch size is 32. To prevent overfitting of the model, a dropout layer is added after the LSTM and GRU layers respectively, and the dropout rate is 0.5.

| **Parameter Setting** | **LSTM** | **GRU** |
| --- | --- | --- |
| Word Vector | 13 | 13 |
| Unit | 50 | 50 |
| Epoch | 2 | 2 |
| Batch Size | 32 | 32 |
| Dropout Rate | 0.5 | 0.5 |

Table 8 Parameter Settings

## Performance Comparison

The evaluation values obtained on the Wikipedia toxic comments dataset for the two models in this paper, LSTM, and GRU, are shown below. Since the type of classification for comments in this project is a combination of several binary classification problems, the evaluation of the model performance is separated into evaluations of the ability to distinguish different categories.

Firstly, the confusion matrix of the different labels of the LSTM model is presented in the form of a heat map, where each row represents a label for the classification, and each column from left to right represents the TP, FP, FN, and TN in percentage form, respectively, for the final model evaluation.

图表

描述已自动生成

Figure 12 Confusion Matrix of LSTM

Next, the results of the LSTM model to distinguish comments between the different labels on the test set are then shown in the figure below, with the results including Accuracy, Precision, Recall, and the F1-score for each category. Support refers to the number of samples in the test set for each label, macro avg refers to the arithmetic mean of the Precision, Recall, and F1-score of both categories in one label, and weighted avg refers to the weighted mean of the Precision, Recall, and F1-score of both categories, where the weight of each category is determined by the number of samples in the test set. Overall, the accuracy of predicting whether a text is toxic is 92%, and the accuracy of determining the label of toxic comments is 97.72% on average.

手机屏幕截图

描述已自动生成

Figure 13 Evaluation of LSTM

As can be seen from the table, the model did not predict well for the "threat" and "severe\_toxic" labels, and Precision and Recall were low. The table below shows some of the misclassified comments with their actual label values.

|  | **Comment Text** | **Actual Label** | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Toxicity** | **Severe Toxic** | **Obscene** | **Threat** | **Insult** | **Identity Hate** |
| A | Azerbaijanis Proud of Ramil Safarov If I had the same opportunity under same circumstances, i would kill both bastard and mother fucker armenians bitches soldiers for insulting Azerbaijani flag. Bastard gurgen lay in hell mother fucker. I fuck your soul and your whore mother. Assholes. Fuck armenian fascists and natinoalists. | 1 | 0 | 1 | 1 | 1 | 1 |
| B | fuck islam , fuck Saud -Arabia | 1 | 1 | 1 | 0 | 1 | 1 |

Table 9 Error Examples for LSTM

Second, the predicted labels in the LSTM model and the comment text after the pre-processing stage are presented.

|  | **Comment Text** | | **Predicted Label** | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Original** | **Pre-processed** | **Toxicity** | **Severe Toxic** | **Obscene** | **Threat** | **Insult** | **Identity Hate** |
| A | Azerbaijanis Proud of Ramil Safarov If I had the same opportunity under same circumstances, i would kill both bastard and mother fucker armenians bitches soldiers for insulting Azerbaijani flag. Bastard gurgen lay in hell mother fucker. I fuck your soul and your whore mother. Assholes. Fuck armenian fascists and natinoalists. | azerbaijanis proud ramil safarov opportunity circumstances would kill bastard mother fucker armenians bitches soldiers insulting azerbaijani flag bastard gurgen lay hell mother fucker fuck soul whore mother assholes fuck armenian fascists natinoalists | 1 | 0 | 1 | 0 | 1 | 0 |
| B | fuck islam , fuck Saud -Arabia | fuck islam fuck saud arabia | 1 | 0 | 1 | 0 | 0 | 0 |

Table 10 Error Analysis for LSTM

These were malicious comments that were predicted normal, the first of which was a comment involving obscene, threat, insult, and identity hate, while the second was a comment involving severe toxic, obscene, insult and identity hate. These two comments were subjected to a data pre-processing stage and the textual content generated was similarly displayed. As can be seen from the table, upper- and lower-case letters were standardized in the original text, stop words such as personal pronouns, adverbs, and punctuation were removed, and all real words were retained intact. In the model's prediction of Comment A, although the pre-processed text contains obvious threatening and racist words, such as "kill" and "americans bitches", the LSTM model did not categorize the content into the threat and identity hate label. In the model's prediction of Comment B, the comments were not labeled as insult and identify hate, despite the pre-processed text including insulting and racist phrases such as “fuck isalm”.

The above errors may be attributed to the small number of positive samples for both labels in the training set, which makes it difficult to obtain enough information to accurately distinguish between these categories when training the model.

The following is the evaluation matrix of the GRU model for predicting comment types.

图表

描述已自动生成

Figure 14 Confusion Matrix of GRU

The outcomes of the GRU model to separate comments between the various labels are then displayed in the table below. GRU model can predict whether a comment is an offending comment with the accuracy of 93.78%, and the accuracy of determining the text category reached 97.83% on average.

一些文字和图片的手机截图

描述已自动生成

Figure 15 Evaluation value of GRU

As can be seen from the classification reports above, the GRU model does not classify the severe toxic and threat and identity hate tags well, and although the model is effective in correctly tagging normal comments, it is poor at correctly classifying the above three tags for toxic comments.

|  | **Comment Text** | **Actual Label** | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Toxicity** | **Severe Toxic** | **Obscene** | **Threat** | **Insult** | **Identity Hate** |
| C | EAT SHIT AND DIE YOU FAGGOT NAZI LOVING JEW HATING MOTHERFUCKER!!! I'LL MURDER YOU IN YOUR OWN HOME YOU FAGGOT!! I WILL FIND YOU, SLIT YOUR THROAT, DRINK YOUR BLOOD, PISS DOWN YOUR MOUTH, CUT YOUR LIVER OUT, COOK IT AND EAT IT FOR BREAKFAST, LUNCH AND DINER. THEN I WILL FUCK YOUR MOM, DIG YOUR GRANDMOM UP FROM HER GRAVE AND FUCK HER IN HER ASS LIKE SHE COULDN'T GET IT FROM YOUR GRANDDAD. I'LL DIG HIM UP TO, SIT HIM UP AND HE CAN WATCH ME ASS FUCK YOUR DEAD GRANNY! | 1 | 1 | 1 | 1 | 1 | 1 |
| D | FUCK YOU YOU LIKE ANGELIQUE CARRINGTON AND TAMARA HARRINGTON??????FUCK YOU ASSHOLE!!!!!! | 1 | 1 | 1 | 0 | 1 | 0 |

Table 11 Error Examples of GRU

The table above shows some of the misclassified comments with their actual label values. These were malicious comments that were predicted to be toxic, however categories were incomplete. The first of which was a comment extremely toxic, involving all the labels provided, while the second was a comment involving severe toxic, obscene, and identity hate.

|  | **Comment Text** | | **Predicted Label** | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Original** | **Pre-processed** | **Toxicity** | **Severe Toxic** | **Obscene** | **Threat** | **Insult** | **Identity Hate** |
| C | EAT SHIT AND DIE YOU FAGGOT NAZI LOVING JEW HATING MOTHERFUCKER!!! I'LL MURDER YOU IN YOUR OWN HOME YOU FAGGOT!! I WILL FIND YOU, SLIT YOUR THROAT, DRINK YOUR BLOOD, PISS DOWN YOUR MOUTH, CUT YOUR LIVER OUT, COOK IT AND EAT IT FOR BREAKFAST, LUNCH AND DINER. THEN I WILL FUCK YOUR MOM, DIG YOUR GRANDMOM UP FROM HER GRAVE AND FUCK HER IN HER ASS LIKE SHE COULDN'T GET IT FROM YOUR GRANDDAD. I'LL DIG HIM UP TO, SIT HIM UP AND HE CAN WATCH ME ASS FUCK YOUR DEAD GRANNY! | eat shit die faggot nazi loving jew hating motherfucker i'll murder home faggot find slit throat drink blood piss mouth cut liver cook eat breakfast lunch diner fuck mom dig grandmom grave fuck ass like get granddad i'll dig sit watch ass fuck dead granny | 1 | 0 | 1 | 0 | 1 | 0 |
| D | FUCK YOU YOU LIKE ANGELIQUE CARRINGTON AND TAMARA HARRINGTON??????FUCK YOU ASSHOLE!!!!!! | fuck like angelique carrington tamara harringtonfuck asshole | 1 | 0 | 1 | 0 | 1 | 0 |

Table 12 Error Analysis of GRU

These two comments were subjected to a data pre-processing stage and the textual content generated was similarly displayed. As can be seen from the table, upper- and lower-case letters were standardized in the original text, stop words such as personal pronouns, adverbs and punctuation were removed, and all real words were retained intact. The two comments underwent a data pre-processing phase where the resulting textual content was presented in a comparable manner. The table shows that the original text was standardized to use consistent upper- and lower-case letters, eliminate stop words such as personal pronouns, adverbs, and punctuation, and preserve all real words. After preprocessing, Comment C retained severe toxic vocabulary such as "eat shit", "motherfucker", and phrases with threatening implications such as "slit throat", as well as the word "nazi" which contains discriminatory connotations, but the model did not classify this comment into any of these three labels. Similarly, the preprocessed text of Comment D was also severe toxic, but the model was not able to detect it.

The following table summarizes the performance of the LSTM model and the GRU model with the same hyperparameters setting in this project.

| **Model** | **Category** | **Accuracy** | **Precision** | **Recall** | **F1-Score** |
| --- | --- | --- | --- | --- | --- |
| LSTM | Toxicity | 93% | 91% | 92% | 90% |
| Severe Toxic | 99% | 99% | 99% | 99% |
| Obscene | 95% | 94% | 95% | 94% |
| Threat | 100% | 99% | 100% | 100% |
| Insult | 95% | 94% | 95% | 94% |
| Identity Hate | 99% | 98% | 99% | 98% |
| GRU | Toxicity | 93% | 92% | 93% | 93% |
| Severe Toxic | 99% | 99% | 99% | 99% |
| Obscene | 96% | 95% | 96% | 95% |
| Threat | 100% | 99% | 100% | 100% |
| Insult | 95% | 95% | 95% | 95% |
| Identity Hate | 99% | 98% | 99% | 98% |

Table 13 Model Comparison

From the table, it is clear that the LSTM and GRU are both efficient models with respectable levels of accuracy, precision, recall, and F1 values in categorizing toxic comments across a range of labels. For all categories classified, severe\_toxic was the most accurate, and threat was the least accurate. However, overall, the accuracy of discriminating among all categories was high.

The performance of the GRU model in determining whether the text is toxic and in determining the violated text category is significantly improved compared with the LSTM model. In summary, the Accuracy of GRU can be improved by 0.81%, Precision by 0.53%, Recall by 0.81%, and F1-Score by 0.80% with the same hyperparameters.

However, as the criteria in the table are weighted averages, a very complete presentation of the model's ability to discriminate between labels cannot be made. What is presented in the table also differs from the results of the previous classification report. The reason for this result may be that there is an imbalance between the different labels in the training set. For example, the number of samples in the training set for the toxic comment labeled as threat is an order of magnitude different compared to the other labels, while the other categories have a larger number of samples. As a result, models did not get enough representative samples during training, which may lead to inaccurate prediction of the rare categories during the testing phase. Another possible reason for this is that the number of each category in the test set may vary. Comments labeled as severe\_toxic and threat with identity hate were so small that even if the model predicted them all as negative categories, their accuracy might be high. For the other categories, their accuracy may be lower than that of the first three labels because they have a much larger sample size. In addition, the textual features of the toxicity, obscene and insult label may be more clearly defined, making it easier for the model to distinguish between these categories.

# Professional Issues

## Project Management

### Activities

All the activities related to the objectives are shown in the table below.

| **Objectives** | **Activity** | **Completed** |
| --- | --- | --- |
| 1. Conduct background research on text classification, understand the field and the corresponding technologies. | A1.1 Identify subject keywords | Completed |
| A1.2 Search for relevant essays | Completed |
| A1.3 Read the relevant literature | Completed |
| A1.4 Summarize the advantages and limitations of different technologies | Completed |
| A1.5 Perform a literature review | Completed |
| 2. Collect usable dataset from the Internet. | A2.1 Search for social media comment datasets on Kaggle | Completed |
| A2.2 Download the datasets | Completed |
| A2.3 Identify the structure of the datasets | Completed |
| 3. Clean and pre-process the data for modeling. | A3.1 Search for methods to clean the data | Completed |
| A3.2 Apply methods on datasets | Completed |
| A3.3 Evaluate the process | Completed |
| 4. Extract features from the text in the cleaned datasets. | A4.1 Search for methods to extract data | Completed |
| A4.2 Implement methods on the datasets | Completed |
| A4.3 Evaluate the feature-extracting methods | Completed |
| 5. Train different models using datasets and assess the quality of the models | A5.1 Search for documentation of different models | Completed |
| A5.2 Apply models in Python language | Completed |
| A5.3 Adjust the parameters until the model performs optimally | Completed |
| 6. Analyze the quality of the models and compare the strengths and weaknesses of each model | A6.1 Search for different methods to evaluate the model | Completed |
| A6.2 Apply multiple rubrics to different models | Completed |
| A6.3 Put the results into a table | Completed |
| 7. Develop data and model testing and evaluation strategy | A7.1Set up a test plan for the dataset and model | Completed |
| A7.2 Put the test plan on the dataset and model | Completed |
| A7.3 Evaluate how well the data and model perform | Completed |
| 8. Analyze risk based on current progress | A8.1 Search for risk analysis for deep learning projects | Completed |
| A8.2 Modify the risk analysis based on the used models | Completed |
| A8.3 Implement the analysis of the models | Completed |

Table 14 Activities of Objects

### Schedule

The project plan with a specific timeline and the schedule for the accomplished work is shown below. The green color represents the period of accomplished work.

| **Task** | **Start Date** | **End Date** | **Duration(days)** |
| --- | --- | --- | --- |
| Choose the Topic | 2022/10/16 | 2022/10/25 | 9 |
| Registration Form and Ethics Form | 2022/10/18 | 2022/10/30 | 12 |
| Prepare Project Proposal | 2022/10/16 | 2022/11/5 | 20 |
| Literature Reading | 2022/10/16 | 2022/11/5 | 20 |
| Model Learning | 2022/10/25 | 2022/11/17 | 23 |
| Implementation of Models | 2022/11/17 | 2023/2/12 | 87 |
| Literature Review | 2022/10/17 | 2022/12/5 | 49 |
| Prepare Interim Report | 2022/11/17 | 2022/12/15 | 28 |
| Testing and Evaluation | 2023/1/3 | 2023/3/18 | 74 |
| Prepare Final Report | 2022/12/31 | 2023/4/7 | 97 |
| Create Poster | 2023/4/7 | 2023/4/12 | 5 |

Table 15 Project Plan

图表, 日程表

描述已自动生成

Figure 16 Schedule of the project period

### Project Data Management

Whenever a new version of the code is updated, it will be uploaded to the Baidu Cloud in order to keep track of all project progress.

Whenever a new version of the report of the project or related electronic documentation is updated, it will be uploaded to the Feishu Shared Space in order to keep track of all project progress.

### Project Deliverables

Throughout the execution of this project, the following items are submitted for assessment:

* Project proposal with ethical forms, showing a detailed description of the work to be done. (Submitted)
* Project weekly report containing planned objectives for each week. (Submitted)
* Progress report providing justification of the project. (Submitted)
* Project presentation illustrated by a poster and a practical demonstration. (Submitted)
* The final report comprising a complete and clear explanation of the problem to be solved. (Submitted)

## Risk Analysis

Possible risks that may appear through the process of accomplishing the project with possible mitigation measures are listed below.

Firstly, the level of risk of an event occurring is shown represented by the different colors. The likelihood of occurrence of the risk is divided into six levels, with the most likely risk score being 6, and the probability of occurrence decreasing in descending order. The severity of the risk is divided into five levels.

The risk score is obtained by multiplying the score of the likelihood of occurrence and severity, meaning that risks with higher scores have a greater impact on the entire project. The scoring criteria for risks in the table are as follows. The score of risk in the table is the product of likelihood and severity. A risk score below 6 is considered Acceptable (ACC), a score between 7 and 15 is evaluated as Require Further Investigation (RFI), while a score beyond 15 is defined as Not Acceptable (N/ACC), with red representing Critical Risk, yellow representing Medium Risk and green representing Low Risk.

|  | Negligible | Minor | Moderate | Significant | Severe |
| --- | --- | --- | --- | --- | --- |
| Frequent | 6 | 12 | 18 | 24 | 30 |
| Probable | 5 | 10 | 15 | 20 | 25 |
| Occasional | 4 | 8 | 12 | 16 | 20 |
| Remote | 3 | 6 | 9 | 12 | 15 |
| Improbable | 2 | 4 | 6 | 8 | 10 |
| incredible | 1 | 2 | 3 | 4 | 5 |

Table 16 Risk Severity Matrix

The below table is arranged as:

Potential Risk: the possible risks that may appear in the process of the project

Potential Causes: The reason for having the risk

Severity: The impact degree potential causes may influence the project

Likelihood: The probability of the situation happening

Risk: The score of the Potential Causes

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Risk ID** | **Potential Risk** | **Cause ID** | **Potential Causes** | **Severity** | **Likelihood** | **Risk** |
| R1.1 | Missed deadline | C1.1.1 | Illness | 1 | 7 | 7 |
| C1.1.2 | Cannot choose the topic | 1 | 1 | 1 |
| C1.1.3 | Poor time management | 6 | 3 | 18 |
| R1.2 | Feature creep | C.1.2.1 | Spend much time on one specific chapter | 3 | 2 | 6 |
| R1.3 | Software bugs | C1.3.1 | Inconsiderate design | 1 | 3 | 3 |
| C1.3.2 | Poor test plan | 4 | 3 | 12 |
| R1.4 | Loss of data | C1.4.1 | Poor version control | 4 | 4 | 16 |
| C1.4.2 | Poor data management | 4 | 4 | 16 |

Table 17 Risk with Potential Causes

The below table illustrates the risks with strategies that may prevent risk from happening.

|  |  |  |  |
| --- | --- | --- | --- |
| **Risk ID** | **Cause ID** | **Mitigation ID** | **Mitigation** |
| R1.1 | C1.1.1 | M1.1.1 | Inform the supervisor in time |
| C1.1.2 | M1.1.2 | Conduct research early and meet the supervisor to discuss |
| C1.1.3 | M1.1.3 | Follow the Gantt Chart strictly |
| R1.2 | C.1.2.1 | M1.2.1 | Discuss a plan with the supervisor early. Create basic goals and enhancements. |
| R1.3 | C1.3.1 | M1.3.1 | Create a highly modular design before implementation |
| C1.3.2 | M1.3.2 | Create a test plan at the start and be ready to change |
| R1.4 | C1.4.1 | M1.4.1 | Implement a version control strategy at the start. |
| C1.4.2 | M1.4.2 | Post every change of the code into GitHub on time |

Table 18 Mitigating Measures

Based on the above diagram, it is clear that the most significant risks to the project are delays due to the lack of time management and data loss due to the lack of management of project versions and data. These two risks can lead directly to the non-delivery of the project, resulting in serious consequences. These risks can be prevented by strict adherence to the Gantt chart and by the development of a project management strategy in advance.

## Professional Issues

### Legal Issue

This project strictly follows the regulations outlined in the General Data Protection Regulation (GDPR) [31] and is used to ensure that the data used for training and testing is legitimate, particularly with regard to transparency and purpose limitation. According to the GDPR, personal data must be processed in a lawful, fair, and transparent manner and can only be used for specific and lawful purposes. This means that the collection and use of user data for sentiment analysis are clearly stated and the consent of the user is obtained before the data is collected. In addition, the user whose information is collected has the right to access his or her data and to request its deletion.

### Ethical Issue

The ethical issue raised by this project is that the model's accuracy may not be reliable when applied to real social media platforms due to the constant evolution of language and new online terminology in today's online society. Although the model may perform well on the data it was trained on, it may struggle to accurately interpret the newly emerging language or cultural references, leading to misjudgments and potentially unfair consequences. For instance, non-malicious comments may be mistakenly flagged as violating community guidelines, resulting in account bans or other penalties for users who have not actually broken any rules. To address this ethical issue, a continuous update has added to the model's training data to incorporate new language and cultural references as they emerge. Additionally, mechanisms for human review and oversight are built in to ensure that the model's decisions are not causing harm or unfairly penalizing users.

### Social Issue

The use of deep learning models in this project raises concerns about algorithmic transparency. Specifically, the black-box nature of the model makes it difficult to understand how the system arrives at its decisions. Without a clear understanding of the decision-making process, it is challenging to ensure that the model is making unbiased and ethical decisions. In addition, the lack of transparency can make it difficult to identify and fix any errors or biases in the model. For example, if the model is misclassifying certain comments as inappropriate, it may be challenging to identify the root cause of the issue without a clear understanding of how the model arrived at that decision.

### Environment Issue

The main purpose of this model is to save human and time resources on social media platforms by automatically screening malicious comments by machines, which is environmentally friendly and greatly reduces the consumption of resources.

The implementation of this model requires a significant number of computational resources, which can contribute to environmental issues such as global warming. The training of deep learning models often involves the use of large amounts of data and can require high-performance computing systems that consume a significant amount of energy. This can result in a high carbon footprint and contribute to global warming. Additionally, the production and disposal of electronic devices, such as GPU and other computer components, can also have a negative impact on the environment.

# Conclusion

## Conclusion and Reflection

Due to the continuous deterioration of the online environment, malicious speech appearing on public social platforms needs to be screened and blocked. However, as manual review of malicious comments consumes excessive social resources, this paper proposes a deep learning-based solution that can effectively reduce the waste of resources. This paper focuses on text classification based on two different RNN variant models on users' comments on the Internet. The experiments use the same training and testing sets extracted from Wikipedia Talk Page on two different deep learning models, LSTM and GRU, to compare the effectiveness of the two models in classifying malicious text. The results show that both deep learning models perform well on the Wikipedia dataset, recognizing and classifying the sentiment of text with high accuracy, but both models not perform satisfactorily on the “Threat” and “Severe toxic” types of text. The reasons for this result could be an imbalance between the different labels in the training set, or a lack of clarity in the text features of these two labels. On a general level, when evaluating the results of the two models, it is found that the model using the GRU method performed better than the model using the LSTM method, with higher Accuracy, Recall and F1-Score.

## Future Work

Future research should concentrate on ways to increase the precision of models. The accuracy of models can be increased by using attentional mechanisms to improve semantic understanding. The following areas will be the focus of follow-up work:

1. Employing attention mechanisms to enhance the model further. Attention mechanisms can help the model to focus on specific parts of the input data and learn important features that are relevant to the task at hand. The problem of lacking clarity of text features of some specific labels may be tackled through the implementation.
2. Exploring more complex LSTM structures or other RNN structures to enhance the model's performance and accuracy. This could include incorporating additional layers into the LSTM architecture, such as stacked LSTMs or a combination of LSTM and CNN layers. Additionally, other types of RNN structures could be investigated, such as the bidirectional LSTM. The bidirectional LSTM allows the model to process both past and future context, providing a richer representation of the input sequence. Exploring these more intricate structures may help the model better capture the underlying relationships and patterns in the text data, improving performance and accuracy.
3. Developing a new attention mechanism and network architecture. The multi-label classifier is a specialized network structure that is capable of handling multiple labels for a single input instance. This can be accomplished using a variety of techniques, including the use of hierarchical classification algorithms or the incorporation of numerous output layers and loss functions. The model may be better able to manage the complexity and variety of multi-label classification problems by leveraging these cutting-edge neural network architectures, leading to enhanced accuracy and performance. To ensure the efficacy and robustness of the model, comprehensive evaluation and validation and neural network design should be based on the specific requirements and characteristics of the toxic comment dataset.

# References

[1] N. Fan, Y. An, and H. Li, “Research on Analyzing Sentiment of Texts Based on K-nearest Neighbor Algorithm,” *Computer Engineering and Design*, vol. 33, no. 3, pp. 1160–1164, 2012, doi: 10.16208/j.issn1000-7024.2012.03.053.

[2] C. Shang, M. Li, S. Feng, Q. Jiang, and J. Fan, “Feature selection via maximizing global information gain for text classification,” *Knowl Based Syst*, vol. 54, pp. 298–309, 2013, doi: 10.1016/j.knosys.2013.09.019.

[3] Li R, “Research on Text Classification and Its Related Technologies,” Apr. 2005.

[4] A. Dhar, H. Mukherjee, N. S. Dash, and K. Roy, “Text categorization: past and present,” *Artif Intell Rev*, vol. 54, no. 4, pp. 3007–3054, Apr. 2021, doi: 10.1007/s10462-020-09919-1.

[5] J. S. Su, B. F. Zhang, and X. Xu, “Advances in machine learning based text categorization,” *Ruan Jian Xue Bao/Journal of Software*, vol. 17, no. 9, pp. 1848–1859, Sep. 2006, doi: 10.1360/jos171848.

[6] G. Wei and K. Wu, “Sentiment Analysis Based on Word Vector Model,” vol. 26, no. 3, 2016, doi: 10.15888/j.cnki.csa.005655.

[7] T. Tang, X. Tang, and T. Yuan, “Fine-Tuning BERT for Multi-Label Sentiment Analysis in Unbalanced Code-Switching Text,” *IEEE Access*, vol. 8, pp. 193248–193256, 2020, doi: 10.1109/ACCESS.2020.3030468.

[8] M. Schuster and K. K. Paliwal, “Bidirectional recurrent neural networks,” *IEEE Transactions on Signal Processing*, vol. 45, no. 11, pp. 2673–2681, 1997, doi: 10.1109/78.650093.

[9] Y. Shen, S. Tan, A. Sordoni, and A. Courville, “Ordered Neurons: Integrating Tree Structures into Recurrent Neural Networks,” Oct. 2018.

[10] Ding Y, “Research on Multi-granular Text Classification of Hate Speech and Abusive Language based on RNN.” https://www.cnki.net (accessed Jan. 09, 2023).

[11] G. Liu and J. Guo, “Bidirectional LSTM with attention mechanism and convolutional layer for text classification,” *Neurocomputing*, vol. 337, pp. 325–338, Apr. 2019, doi: 10.1016/j.neucom.2019.01.078.

[12] H. Khataei Maragheh, F. S. Gharehchopogh, K. Majidzadeh, and A. B. Sangar, “A New Hybrid Based on Long Short-Term Memory Network with Spotted Hyena Optimization Algorithm for Multi-Label Text Classification,” *Mathematics*, vol. 10, no. 3, p. 488, Feb. 2022, doi: 10.3390/math10030488.

[13] A. Onan and M. A. Tocoglu, “A Term Weighted Neural Language Model and Stacked Bidirectional LSTM Based Framework for Sarcasm Identification,” *IEEE Access*, vol. 9, pp. 7701–7722, 2021, doi: 10.1109/ACCESS.2021.3049734.

[14] R. K. Behera, M. Jena, S. K. Rath, and S. Misra, “Co-LSTM: Convolutional LSTM model for sentiment analysis in social big data,” *Inf Process Manag*, vol. 58, no. 1, p. 102435, Jan. 2021, doi: 10.1016/j.ipm.2020.102435.

[15] “Wikipedia:About.” http://en.wikipedia.org/wiki/Wikipedia:About (accessed Mar. 28, 2023).

[16] “Kaggle: Toxic Comment Classification Challenge.” https://www.kaggle.com/competitions/jigsaw-toxic-comment-classification-challenge (accessed Mar. 28, 2023).

[17] H. Jelodar, Y. Wang, R. Orji, and S. Huang, “Deep Sentiment Classification and Topic Discovery on Novel Coronavirus or COVID-19 Online Discussions: NLP Using LSTM Recurrent Neural Network Approach,” *IEEE J Biomed Health Inform*, vol. 24, no. 10, pp. 2733–2742, Oct. 2020, doi: 10.1109/JBHI.2020.3001216.

[18] H. Li, “A review of natural language processing based on RNN and Transformer models,” *Information Recording Material*, vol. 22, no. 12, pp. 7–10, 2021, doi: 10.16009/j.cnki.cn13-1295/tq.2021.12.081.

[19] Y. Xia, “A Review of the Development of Recurrent Neural Network,” *Computer Knowledge and Technology*, vol. 15, pp. 182–184, Jul. 2019, doi: 10.14004/j.cnki.ckt.2019.2379.

[20] S. Hochreiter and J. Schmidhuber, “Long Short-Term Memory,” *Neural Comput*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997, doi: 10.1162/neco.1997.9.8.1735.

[21] J. Koutník, K. Greff, F. Gomez, and J. Schmidhuber, “A Clockwork RNN,” Feb. 2014, Accessed: Jan. 08, 2023. [Online]. Available: http://arxiv.org/abs/1402.3511

[22] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, “Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling,” Dec. 2014.

[23] J. Fang, P. Chen, and W. Liao, “Text Classification Model Based on GloVe and GRU”, doi: 10.3778/j.issn.1002-8331.2001-0272.

[24] A. Vaswani *et al.*, “Attention Is All You Need,” Jun. 2017.

[25] S. Wang and C. D. Manning, “Baselines and Bigrams: Simple, Good Sentiment and Topic Classification,” 2012. [Online]. Available: http://www.cs.uic.edu/

[26] X. Tang, J. Tang, and A. Zhou, “Research on Community Malicious Comment Detection based on a Hybrid Model of Feature Selection and Random Forest,” *Morden Computer*, vol. 2020, no. 07, pp. 22–26, 2020, doi: 10.3969/j. issn. 1007-1423. 2020. 19. 005.

[27] Pratap D, “Toxic Comment Glove And GRU with 97% Accuracy.” https://www.kaggle.com/code/dhirendra73/toxic-comment-glove-and-gru-with-97-accuracy (accessed Mar. 20, 2023).

[28] Afzal B, “Toxic Comments Classification using Transformers.” https://www.kaggle.com/code/bilalafzal255/toxic-comments-classification-using-transformers (accessed Mar. 20, 2023).

[29] Zhu D, “Random Forest Classification.” https://www.kaggle.com/code/grandmoon/random-forest-classification (accessed Mar. 20, 2023).

[30] Abhi, “Naive Bayes Baseline and Logistic Regression.” https://www.kaggle.com/code/abhi111/naive-bayes-baseline-and-logistic-regression/notebook (accessed Mar. 20, 2023).

[31] F THE EUROPEAN PARLIAMENT AND OF THE COUNCIL, “General Data Protection Regulation,” *Official Journal of the European Union*, 2016.