الجمهورية الجزائرية الديموقراطية الشعبية

Ministry of Higher Education and Scientific Research

Yahia Fares University of Medea

Faculty of Sciences

Department of Mathematics and Computer Science



Graduation Project

Submitted for Obtaining the Master's Degree in Computer Science

Speciality: Engineering Systems and Web Technology

Leveraging Large Language Models for Cyberbullying Detection

Proposed and directed by:

Dr. Azzeddine Benaissa Rachid

Prepared and presented by:

Mr. Boumenir Ismail

Mr. Benyahia Fateh

2024-2025

Acknowledgement

In the name of Allah, the Most Kind and Merciful.

We want to thank **Dr. Azzeddine Benaissa Rachid**, our supervisor, from the bottom of our hearts for all the help, support, and encouragement he gave us during this project. His knowledge and comments have been very helpful in guiding the work's direction and quality.

We would also like to thank the **Department of Mathematics and Computer Science** at **Yahia Fares University** of Medea for giving us the academic setting, tools, and help we needed to finish this project.

We are very grateful to our families for their tolerance, understanding, and continual support, which have helped us get through tough times. We will always be thankful to our parents for their sacrifices, prayers, and faith in us.

We also want to thank our friends and classmates who went through this academic adventure with us. Your help, encouragement, and moral support were a big part of what made this experience so valuable and memorable.

Lastly, we thank Allah (SWT) for granting us strength, willpower, and health to finish this thesis. He is the source of all success.

Dedication

To my parents,
who gave me roots to hold me to earth and wings to take me to high skies—
your love has served as guideposts to me, wisdom shining like shining star.
To all all-nighter conversations, all prayers in secret, all "I believe in you"
which kept me going through every storm, part of this work is in way of expression:
Your believing in me constructed mountains on which now I walk.

To friends,
fellow mess travelers in messiness and defenders of oddities—
you energized me through coffee, memes, and relentless optimism.
Where journey became rough, you became step stool to me; where I stumbled,
you rolled me around in tumbling stories. This work belongs to us equally to me.

We together taught me that success is more delectable in sharing.
Thank you for coloring my universe with love, determination, and delight.

Forever in gratitude

BENYAHIA Fateh.

Dedication

In the name of Allah, the Most Gracious, the Most Merciful
To my dear family, I thank Allah first for giving me a family like you, then I want to thank you for your continued support during my studies and beyond and for the sacrifices you made for me and for the moral and material support.

especially my father, may Allah have mercy on him, who was not just a father but a friend who was with me every time and whom I wished to be present to see the end of my academic life, but Allah's will and what He wants happens All love and gratitude to you for all your efforts and sacrifices upon me, and I pray to Allah that you remain in good health and well-being.

To my dear friends, who are like brothers to me, and especially those who shared with me the academic path, thank you for the moral support and for putting efforts in the various projects we worked on,
and for the beautiful memories we shared together throughout the academic path and for the difficulties we overcame by the blessings of Allah,
I pray to Allah for success for me and for you in your lives and I pray to Allah for you to be healthy and well

BOUMENIR Ismail.

Abstract

This work explores the application of Large Language Models (LLMs), such as LLaMA and Mistral, in advancing cyberbullying detection. Unlike traditional rule-based approaches, LLMs leverage contextual understanding and emotional cues to identify nuanced forms of online abuse, including sarcasm, coded language, and informal dialects. The study employs fine-tuned models adapted to annotated datasets, utilizing multitask learning for both hate classification and intent detection. Key challenges addressed include cultural bias, privacy risks, and the unpredictable nature of online language. Results indicate that LLMs significantly enhance detection accuracy and adaptability. Future directions involve integrating multimodal data (e.g., text with images or audio), leveraging federated learning for privacy preservation, and aligning detection systems with ethical AI frameworks to ensure fairness and user safety. This approach highlights the potential of LLMs to enable proactive and responsible cyberbullying mitigation on digital platforms.

Keywords: LLMs, NLP, cyberbullying detection, contextual understanding, bias mitigation, federated learning, ethical AI

Résumé

Cette étude explore l'application de grands modèles linguistiques (LLM), tels que LLaMA et Mistral, pour améliorer la détection du cyberharcèlement. Contrairement aux approches traditionnelles basées sur des règles, les LLM s'appuient sur la compréhension contextuelle et les signaux émotionnels pour identifier les formes nuancées d'abus en ligne, notamment le sarcasme, le langage codé et les dialectes informels. L'étude utilise des modèles affinés, adaptés à des ensembles de données annotées, utilisant l'apprentissage multitâche pour la classification des propos haineux et la détection des intentions. Les principaux défis abordés comprennent les préjugés culturels, les risques pour la vie privée et la nature imprévisible du langage en ligne. Les résultats indiquent que les LLM améliorent considérablement la précision et l'adaptabilité de la détection. Les orientations futures incluent l'intégration de données multimodales (par exemple, texte avec images ou audio), l'exploitation de l'apprentissage fédéré pour la préservation de la confidentialité et l'alignement des systèmes de détection sur les cadres éthiques de l'IA afin de garantir l'équité et la sécurité des utilisateurs. Cette approche met en évidence le potentiel des LLM pour permettre une atténuation proactive et responsable du cyberharcèlement sur les plateformes numériques.

Mots-clés: Grand modéle de langage(LLMs), NLP (TALN), Cyberharcèlement detection, analyse contextuelle, atténuation des biais, IA éthique.

ملخص

تستكشف هذه الدراسة تطبيق نماذج اللغة الكبيرة(LLMs) ، مثل LLaMA وMistral، لتحسين الكشف عن التنمر الإلكتروني. بخلاف النُهج التقليدية القائمة على القواعد، تعتمد نماذج اللغة الكبيرة على الفهم السياقي والإشارات العاطفية لتحديد الأشكال الدقيقة للإساءة عبر الإنترنت، بما في ذلك السخرية واللغة المشفرة واللهجات غير الرسمية. تستخدم الدراسة نماذج مُحسّنة، مُكيّفة مع مجموعات البيانات المُعلّقة، باستخدام التعلم متعدد المهام لتصنيف خطاب الكراهية وكشف النوايا. تشمل التحديات الرئيسية التي تمت معالجتها التحيز الثقافي، ومخاطر الخصوصية، والطبيعة غير المتوقعة للغة الإنترنت. تشير النتائج إلى أن نماذج اللغة الكبيرة تُحسّن دقة الكشف وقابليته للتكيف بشكل ملحوظ. تشمل التوجهات المستقبلية دمج البيانات متعددة الوسائط (مثل النصوص مع الصور أو الصوت)، والاستفادة من التعلم المُوحّد للحفاظ على الخصوصية، ومواءمة أنظمة الكشف مع الأطر الأخلاقية للذكاء الاصطناعي لضمان العدالة وسلامة المستخدم. يُسلّط هذا النهج الضوء على إمكانات نماذج اللغة الكبيرة في تمكين التخفيف الاستباقي والمسؤول من التنمر الإلكتروني على المنصات الرقمية.

الكلمات المفتاحية: نماذج اللغة الكبيرة(LLMs) ، معالجة اللغة الطبيعية(NLP) ، اكتشاف التنمر الإلكتروني، التحليل السياقي، التخفيف من التحيز، الذكاء الاصطناعي الأخلاقي.

Table Of Contents

List of Figures **4**
List of Tables **6**
Abbreviations **7**
General Introduction **8**
Chapter 1: Cyberbullying Detection Problem **10**
1.1 Introduction 11
1.1.1 Definition of Cyberbullying 11
1.1.2 Background and History of Cyberbullying 11
1.1.3 Importance of Addressing Cyberbullying 12
1.2 Types and Forms of Cyberbullying 12
1.3 Causes and Motivations Behind Cyberbullying 14
1.4 Adverse Effects of Cyberbullying 14
1.4.1 Negative Effects of Cyberbullying 14
1.4.2 Cyberbullying Among College and University Students 15
1.4.3 Cyberbullying of Adults 16
1.5 Legal and Policy Measures Against Cyberbullying 16
1.5.1 Legal and Policy Framework 16
1.5.2 National Responses to Cyberbullying 17
1.5.3 Efforts Across the EU 17
1.6 Previous Solutions to Combat Cyberbullying 17
1.7 Future Strategies to Combat Cyberbullying 20
1.8 Case Studies and Real-Life Examples of Cyberbullying 21
1.9 Modern Challenges in Cyberbullying Prevention 22
1.10 Computer Science in Fighting Cyberbullying 23
1.11 Conclusion 23
Chapter 2: Machine/Deep Learning **24**
2.1 Introduction 25
2.2 Definition of Machine learning 25
2.3 Machine Learning Fundamentals 25
2.3.1 Types of Machine Learning 26
2.3.2 Machine Learning Algorithms : 27

2.3.3 Performance Evaluation Metrics : 29
2.4 Definition of Deep Learning 32
2.5 Deep Learning Fundamentals : 34
2.5.1 Neural Networks and Deep Neural Networks: 34
2.5.2 Importance of Neural Networks : 35
2.5.3 Neural Network Architecture Layers : 36
2.6 The main differences between Machine Learning and Deep Learning 36
2.7 What Is Natural Language Processing(NLP) 37
2.8 Explanation of Large Language Models (LLMs) : 37
2.8.1 Definition of Large Language Models (LLM) 38
2.8.2 Importance of Large Language Models : 39
2.8.3 Large Language Models Examples : 39
2.8.4 How Do Large Language Models Work 40
2.8.5 The Challenges of Large Language Models: 41
2.9 Related Works : 43
2.10 Conclusion 47
Chapter 3: Technical details & Proposed Approach 48
3.1 Introduction 49
3.2 Methodology adopted 49
3.2.1 Used Libraries 49
3.3 Environment and Development tools 51
3.4 Datasets 52
3.4.1 Pre-Processing 53
3.4.2 Splitting the data 53
3.5 Proposed method for unbalanced Datasets : 54
3.6 Models Fine tuning with PEFT (LoRA) 54
3.7 Models used 55
3.8 Conclusion 61
Chapter 4: Results & Discussion 62
4.1 Introduction 63
4.2 Models Training Parameters 63
4.3 Results of Impermium / Formsprimg datasets: 68

4.4 Dataset Comparison Using Weighted Cross-Entropy loss 74
4.5 Dataset Comparison Using Focal Loss 76
4.6 Comparison Between Cross Entropy loss and Focal Loss methods 78
4.7 Comparative Analysis of Best Models Performances with previous Studies 79
4.8 Analysis of the Factors Influencing the Results 81
4.9 Cyberbullying detection User Interface with Fine-Tuned Language Models 82
4.10 Conclusion 83
General Conclusion 84
Bibliography 85

List of Figures

Figure 1.1 Student Experience with Bullying and Cyberbullying 12
Figure 1.2 Types of Cyberbullying 13
Figure 1.3 What Works to Stop Cyberbullying 20
Figure 2.1 Types of Machine Learning 27
Figure 2.2 Confusion Matrix 30
Figure 2.3 Precision 31
Figure 2.4 Recall 31
Figure 2.5 Accuracy 31
Figure 2.6 f1-score 32
Figure 2.7 A Simple Artificial Neural Network 33
Figure 2.8 the analogy between a biological neuron and an artificial neuron, showing how inputs are received and processed to produce outputs in both systems. 35
Figure 2.9 Layers in Neural Network Architecture 36
Figure 2.10 Differences between Machine / Deep Learning 37
Figure 2.11 How transformer models work. 38
Figure 2.12 Image shows the structure of encoder-decoder language models. 41
Figure 2.13 Compute required for training transformer models. 42
Figure 3.1 Fine-Tuning Large Language Models for Cyberbullying Detection 49
Figure 3.2 Formspring and Impermium Class Distributions 53
Figure 3.3 Mistral Model. 55
Figure 3.4 Comparison Mistral 7B with Llama2 Family. 56
Figure 3.5 Llama Model Family. 57
Figure 3.6 Llama 3.2 Family. 57
Figure 3.7 SmolLM2 Model 59
Figure 3.8 Falcon3 Model. 60
Figure 4.1 Model & Quantization Parameters 64
Figure 4.2 LoRA Configuration 65
Figure 4.3 TrainingArguments 66
Figure 4.4 Weighted Cross-Entropy Loss Parameters 67
Figure 4.5 Focal loss parameters 67
Figure 4.6 Confusion Matrices Impermium Dataset in Weighted Cross-Entropy Loss Method 70
Figure 4.7 Confusion Matrices Formspring Dataset in Weighted Cross-Entropy Loss Method 71
Figure 4.8 Confusion Matrices Impermium Dataset in Focal Loss Method 72
Figure 4.9 Confusion Matrices Formspring Dataset in Focal Loss Method 73
Figure 4.10 Recall Comparison Across Dataset in Weighted Cross Entropy Loss Method for Non Cyberbullying class 75
Figure 4.11 Recall Comparison Across Dataset in Weighted Cross Entropy Loss Method for Cyberbullying class 75
Figure 4.12 Recall Comparison Across Dataset in Focal loss Method for Non Cyberbullying class 77
Figure 4.13 Recall Comparison Across Dataset in Focal loss Method for Cyberbullying class 77

Figure 4.14 Non-Cyberbullying Detection example 82
Figure 4.15 Cyberbullying Detection example 83

List of Tables

Table 2.1 Related Works table 46
Table 3.1 Distribution of classes and examples of the datasets used 52
Table 3.2 Splitting the data 53
Table 3.3 Benchmarks by their themes. 56
Table 3.4 Llama 3.2 1B & 3B Benchmarks. 58
Table 3.5 evaluation results of SmolLM2 vs other models 59
Table 4.1 Model & Quantization Parameters 64
Table 4.2 LoRA Configuration 64
Table 4.3 TrainingArguments 66
Table 4.4 Loss Function 67
Table 4.5 Model results on the two datasets Formspring and Impermium in Weighted Cross-Entropy Loss Method 68
Table 4.6 Model results on the two datasets Formspring and Impermium in Focal Loss Method 69
Table 4.7 Results of comparisons with reference models 80

Abbreviations

ACC: Accuracy

API: Application Programming Interface

BERT: Bidirectional Encoder Representations from Transformers

CB: Cyberbullying

CPU: Central Processing Unit

DNN: Deep Neural Network

DL: Deep Learning

EU: European Union

GPT: Generative Pre-training Transformer

GPU: Graphics Processing Unit

KNN: K-Nearest Neighbors

LLM: Large Language Model

LoRA: Low-Rank Adaptation

ML: Machine Learning

NCB: Non-Cyberbullying

NLP: Natural Language Processing

SVM: Support Vector Machine

General Introduction

While online friendships and social connections have gone wild, the problem of cyberbullying has become a menacing concern that cuts across age and social media category. Proper prevention and detection are ever so much more needed now, and LLMs are now viable options to fill the need. These powerful AI models, having been pre-trained with massive amounts of textual information, are more adept at identifying hateful and insensitive remarks from web posts than other models.

GPT, BERT, and Mistral are just some of the vast language models that are part of the system in identifying cyberbullying with deep language comprehension, sentiment, and context. In contrast to other abuse detection systems based on keywords that would otherwise trigger false positives or negatives, LLMs are capable of managing the overall context of the conversation and thus are best positioned to determine whether a conversation is abusive or not. With calibration using data on cyberbullying, the models can be trained to detect the explicit instances of abuse online with verbal threats and explicit insults and the passive means of bullying such as exclusion and manipulation.

One of the most significant strengths of LLMs against cyberbullying detection is that it is able to process enormous amounts of data within seconds, thus it is made possible to react as swiftly as possible in online contexts. Besides that, the models can be trained for various social media platforms with language styles specific to a platform, colloquial, and even multimodal inputs like text and image. Although they hold tremendous potential, there are certain drawbacks to applying LLMs to cyberbullying. Biased training data can lead to biased moderation responses, and malicious attacks through the use of deliberate misspellings or coded language can lead to impaired capacity of the models to flag content as offensive. The computational expense of running large models under real-time moderation is also to be considered.

In spite of such difficulties, ongoing improvement of machine learning methods, such as bias reduction methods, adversarial training, and model optimization, is improving the effectiveness and efficiency of LLMs for cyberbullying detection. With ongoing advancements in AI technology, use of large language models in content moderation platforms can contribute significantly to making online communities safer and more friendly.

Motivation

The proliferation of cyberbullying in our progressively digital society endangers both individual welfare and the integrity of online communities. As our lives increasingly transition to virtual environments, the demand for efficient, scalable, and context-sensitive solutions to address online abuse has become paramount. Large language models (LLMs) such as GPT, BERT, and Mistral present a significant source of optimism. Utilizing the profound linguistic comprehension and contextual insight of these models enables us to transcend basic keyword filters, advancing towards more sophisticated, nuanced, and real-time systems that accurately recognize dangerous behavior.

The impetus for utilizing LLMs in cyberbullying detection transcends mere technological progress; it is fundamentally rooted in human concerns. The focus is on safeguarding vulnerable individuals, cultivating digital spaces that promote safe self-expression, and reinstating faith in social media networks. Each advancement in intelligent and equitable moderation results in fewer victims, more resilient communities, and a digital landscape that embodies humanity's finest qualities instead of its basest instincts.

Chapter 1: Cyberbullying Detection Problem

1.1 Introduction

The internet and social media have transformed communication and contact among people in the modern digital age. Though they offer many benefits, these channels have also fostered negative behaviors. Cyberbullying is a major worry.

1.1.1 Definition of Cyberbullying

Bullying that occurs via digital devices, such as computers, tablets, and cell phones, is known as cyberbullying. Cyberbullying can happen online in social media, forums, or games where users can read, interact with, or share content, as well as through SMS, text messages, and applications. Sending, publishing, or disseminating hurtful, malicious, or deceptive content about another person is known as cyberbullying. It may involve disclosing someone else's private or sensitive information, which could be embarrassing or degrading. Cyberbullying sometimes veers into illegal or criminal activity.[1]

The most common places where cyberbullying occurs are:

- Social Media, such as Facebook, Instagram, Snapchat, and Tik Tok
- Text messaging and messaging apps on mobile or tablet devices
- Instant messaging, direct messaging, and online chatting over the internet
- Online forums, chat rooms, and message boards, such as Reddit
- Email
- Online gaming communities

1.1.2 Background and History of Cyberbullying

Since cyberbullying has existed for a very long period, there is a wealth of background information and history available. Cyberbullying occurs when someone uses technology to hurt another person. "Canadian educator Bill Belsey, founder and president of Bullying.org Canada Incorporated, and leader of anti-bullying efforts, is credited with coining the terms cyberbullying and cyberbully" (cyberbullying). People of all ages, from young teens to adults, can be harmed by this long-standing kind of cyberbullying. Threats, sexual harassment, spreading false information, impersonation to post content that harms a person's reputation, ruins relationships, or causes other problems, making someone the target of mockery or derision in online forums, chat rooms, or websites, posting compromising photos taken without the victim's knowledge or consent, or editing photos to embarrass or impugn them—for example, by pasting the victim's face on a nude body—and eroding someone's confidence in order to deceive them into disclosing secrets or embarrassing information are all examples of cyberbullying. (online harassment). Each of these approaches is an illustration of what is happening in modern society.[2]

[1] https://www.stopbullying.gov/cyberbullying/what-is-it
[2] https://rmatusik.wordpress.com/background-of-cyberbullying/

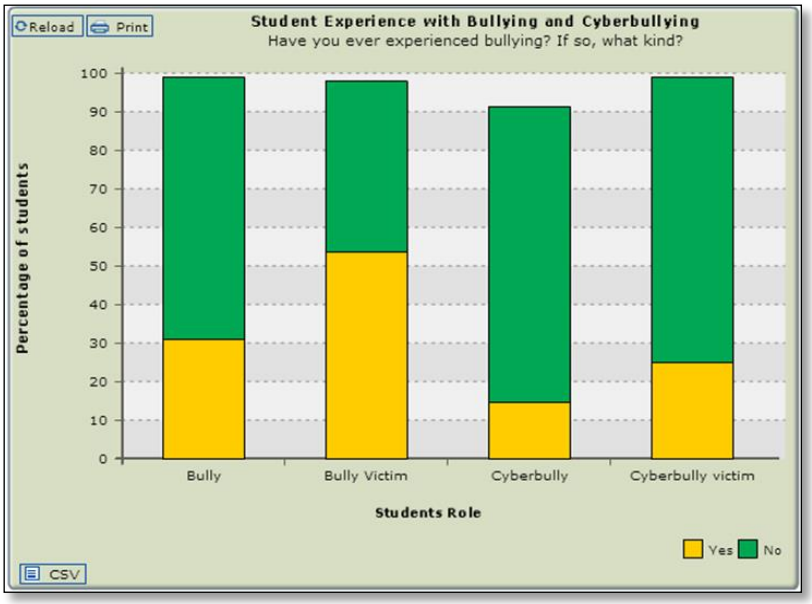


Figure 1.1 Student Experience with Bullying and Cyberbullying

1.1.3 Importance of Addressing Cyberbullying

Cyberbullying and other forms of bullying can have detrimental effects on (communities / organizations / and individuals). A person's emotional and physical health may suffer as a result of cyberbullying. Victims may feel frightened and suffer from severe social isolation. It can result in bodily and emotional injury, low self-esteem, anxiety and humiliation, and trouble focusing and learning. There have also been cases of young persons killing themselves. When someone bullies others, it can lead to continued criminal or antisocial behavior as well as maltreatment in other settings. Whether at a school, workplace, or sports club, cyberbullying is likely to create challenging and harmful environments. It is the duty of the organizations where bullying takes place to deal with and stop bullying. In addition to the serious harm that cyberbullying may cause, failure to address bullying occurrences may put organizations at danger of receiving compensation claims.

1.2 Types and Forms of Cyberbullying

Cyberbullying can take many forms, each with different techniques and effects. The following describes some of the most prevalent types and forms it may take [3]:

Harassment: Sending or disseminating unpleasant, cruel, or abusive messages to a victim over email, social media, or mobile devices is known as online harassment. It also includes purposeful, persistent, and ongoing harassment of the victim through the dissemination of rumors, unwanted or threatening emails, or unpleasant communications.

Fake profiles/Impersonation: Impersonation is the act of posing as someone else in order to damage a victim's or company's reputation by setting up phony email addresses or social media

[3] https://dev.to/shivamchamoli18/what-is-cyber-bullying-44b3

profiles. Getting the victim's photos from the internet and making the account look as real as possible makes it easy to do.

Denigration/Dissing: Denigration occurs when someone shares or broadcasts important details or rumors about the victim in private communications or public posts with the intention of damaging their reputation or interpersonal connections.

Outing/Doxing: Outing is the act of someone publicly disclosing or sharing a victim's private information, photos, or secrets on social media or in public places without the victim's consent or knowledge in order to degrade, disgrace, or shame them.

Exclusion: When someone or a group deliberately keeps the victim out of an online group, this is known as exclusion. It may entail keeping victims out of chat rooms, social media groups, texts, events, or activities.

Cyberstalking: One of the worst forms of cyberbullying is cyberstalking, in which the offender harasses the victim through social media or email. Persistent harassment and defamation are typically used to make the victim feel afraid, but it can also cause bodily harm.

Trickery: Trickery is when someone deceives a victim into disclosing secrets and personal information. It entails winning the trust of victims in order to obtain their secrets, then violating that trust by disclosing private material online.

Flaming: Derogatory remarks about the victim or outright abuse and humiliation are examples of cyberbullying. Similar to trolling, it typically entails a direct attack that uses offensive and furious language to incite the victim to an online altercation.

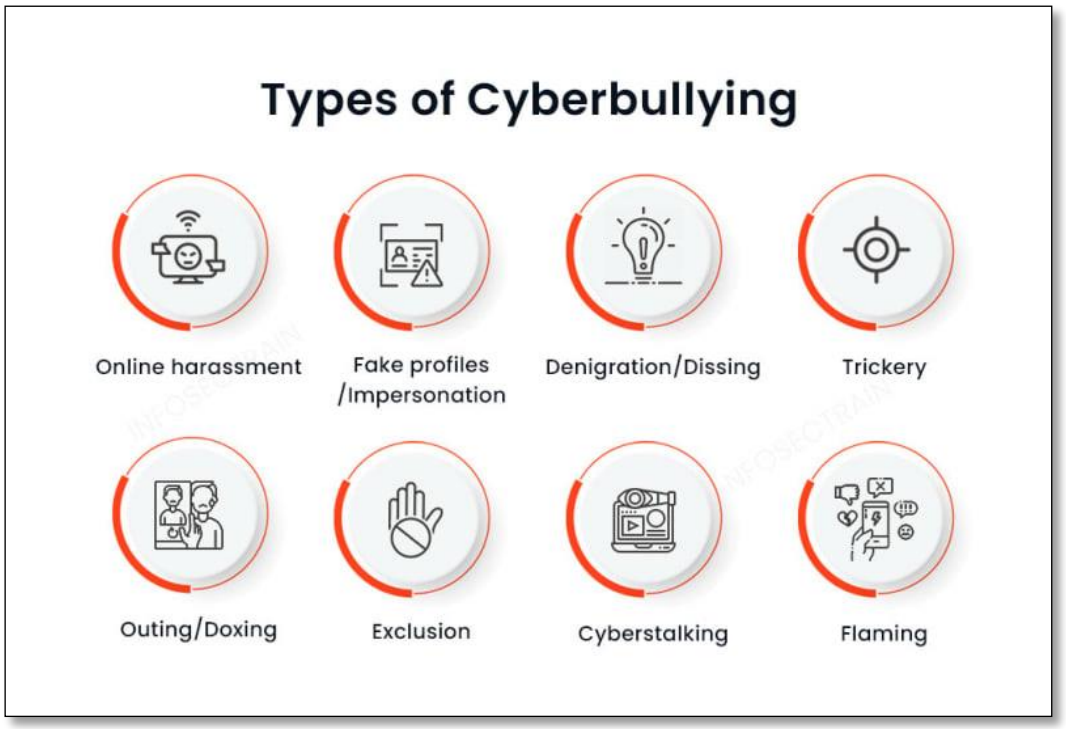


Figure 1.2 Types of Cyberbullying [4]

[4] https://dev.to/shivamchamoli18/what-is-cyber-bullying-44b3

1.3 Causes and Motivations Behind Cyberbullying

Cyberbullying occurs for many of the same reasons as traditional bullying, but because it can be done anonymously, it could be much more alluring. Popular people and those on the periphery of society are the two types of people who are most prone to bully, according to Stopbullying.gov.

Teenagers or popular kids may bully because:

- For them, it's a means of maintaining popularity.
- They feel strong when they hurt other people.

Less socially adept children or teenagers may harass because:

- They may deal with their own poor self-esteem thanks to it.
- They believe it will enable them to blend in with their classmates.
- They find it difficult to sympathize with people they have harmed.

Other reasons why people might engage in online bullying include the following:

Anonymity: Cyberbullying gives the impression that bullies won't be found since it enables them to avoid confronting their victims, requiring less bravery.

Ignorance of the repercussions: According to a National Council on Crime Prevention poll, 81% of students stated they assume others engage in cyberbullying because they find it amusing. Cyberbullies might not be aware of the harm they are causing because they are unable to observe their victims' responses in person.

Social pressure: When friends support them, some cyberbullies may believe that their actions are typical and acceptable in society.

1.4 Adverse Effects of Cyberbullying

Cyberbullying may have serious psychological consequences, including anxiety, depression, and low self-esteem, especially among the youth. In severe cases, repeated exposure to online harassment could cause self-harm or suicidal thoughts, thus highlighting the urgent need for prevention measures and support systems.

1.4.1 Negative Effects of Cyberbullying

A youngster who experiences cyberbullying is more likely to experience bullying at school as well, according to StopBullying.gov. Bullied children are more likely to leave, skip, or quit school than their classmates who are not bullied; those who do not are more likely to experience academic setbacks, including lower grade point averages, standardized test scores, and lower involvement in class.

Numerous detrimental effects that victims of cyberbullying experience include the following, according to research:

- An increase in attempts at suicide
- Increased prevalence of anxiety and sadness
- Diminished sense of self-worth

- More cases of bedwetting, trouble falling asleep, and medical problems like headaches and stomachaches
- Higher rates of eating disorders, which are particularly more common in girls who are in bullying relationships

Some youngsters who have been bullied have retaliated violently. For instance, the shooters in 12 out of 15 school shooting events during the 1990s had experienced bullying themselves.

The victims of bullying are not the only ones harmed. According to StopBullying.gov, offenders are more likely to:

- Engage in combat
- Deface property
- Quit school
- misuse drugs and alcohol during youth and adulthood.
- possess adult traffic tickets and criminal convictions
- Adults who abuse their spouses, children, or intimate partners

1.4.2 Cyberbullying Among College and University Students

When it occurs amongst college students, cyberbullying—also known as electronic victimization—is still not widely acknowledged. However, as college students are among the most frequent consumers of digital technology. this is not an unusual occurrence in that setting. Electronic identity criticism, sexual harassment, and the "outing" of private information like sexual orientation or medical diagnoses (such mental illnesses or STDs) without consent are all examples of cyberbullying in higher education. Although harassment, hazing, intimidation, stalking, and other forms of online hostility between college students have had devastating effects, we don't typically classify the behavior of legal adults as bullying. Of them:

- As of 2014, the University of Wisconsin-River Falls After being harassed online after it was discovered that she had been in an adult video, a student named Alyssa Funke committed suicide.
- In 2010, Tyler Clementi, a freshman at Rutgers University, committed himself by jumping from the George Washington Bridge after learning that his roommate and a friend had secretly recorded an intimate encounter he had with another man and posted it online.

Dr. Jones Peled's study "Cyberbullying and Its Influence on Academic, Social, and Emotional Development of Undergraduate Students" was published by the National Center for Biotechnology Information in 2019. The following consequences of cyberbullying among college students were mentioned:

In order to escape the torture, victims often experience psychological issues like depression, anxiety, loneliness, low self-esteem, social exclusion, school phobias, poor academic performance, family issues, school violence, and delinquent behavior. These issues often lead to suicidal thoughts.
Additionally, studies have demonstrated that cyberbullying harms the emotional and physical health of helpless victims in addition to causing psychosocial issues like improper conduct, alcohol and tobacco usage, despair, and a lack of academic dedication.

Because they are under a lot of emotional stress, victims of cyberbullying find it difficult to focus on their schoolwork, which hinders their academic success. Because victims are frequently psychologically harmed, cyberbullying has a depressing effect that inhibits students from achieving academic success.

1.4.3 Cyberbullying of Adults

Adults are suffering cyberbullying in alarming numbers, despite the common misconception that it only affects kids and students and is merely an online manifestation of playground harassment.

The Pew Research Center conducted a poll in September 2020 to gauge American citizens' experiences with six major types of online harassment. 41% of Americans have directly encountered cyberbullying in at least one of the following types, according to the study9: persistent harassment, sexual harassment, physical threats, stalking, derogatory name-calling, and intentional embarrassment. Of those people asked, 25% said they had experienced at least one of the first four, which are thought to be the more severe of those six. Compared to 15% in 2014 and 18% in 2017, that response rate has increased. Additionally, people who have experienced cyberbullying are more likely to say that their most recent encounter entailed more severe and diversified forms of cyberbullying than they were in 2017.

Adults may suffer from cyberbullying just as much as children and college students do, if not more. The impact of "an internet-based, adult hate campaign can be devastating—cyberbullying, reputation attacks, and harassment online have all seen lives lost, careers and businesses damaged, social and professional reputations destroyed, and entire families broken beyond repair," according to the website of the Cybersmile Foundation, an award-winning nonprofit dedicated to digital wellbeing and combating bullying and abuse online.

1.5 Legal and Policy Measures Against Cyberbullying

Legal and policy measures addressing cyberbullying have increasingly garnered attention at both national and international levels. Although a cohesive legal framework specifically prohibiting cyberbullying is absent across all jurisdictions, numerous techniques and initiatives have been implemented to safeguard individuals—particularly minors—from online threats. This section examines the principal frameworks and legislative initiatives pertaining to the topic.

1.5.1 Legal and Policy Framework

Even though the European Union (EU) and other agencies have broached the issue of cyberbullying, no EU law makes online harassment an offense. However, a number of measures aim at protection of children online, including the UN Convention on the Rights of the Child, the EU Better Internet for Kids (BIK+) Strategy, and the Digital Services Act (DSA). They aim at protecting children from dangerous content, increasing digital literacy, and regulating online platforms. The European Parliament urged stronger EU-wide action against cyberbullying and advocated policies to drive criminal activity online to the same legal level as against offline activity. Recent debates in parliament have focused on applying overall measures to protect children against abuse online. [1]

1.5.2 National Responses to Cyberbullying

Many EU Member States have enacted national policies for addressing cyberbullying, though their legislation varies.

Ireland: There is no single law of cyberbullying, but Ireland's "Coco's Law" makes online harassment and publishing non-consensual photographs a criminal offense. [1]

France: Cyberstalking is a criminal act, punishable by imprisonment and fines, and there are enhanced penalties when the victim is a child. [1]

Italy: Cyberbullying is codified in law in a 2017 act, and offending content may be asked to be taken down by the authorities. Online harassment, defamation, and identity theft are addressed through various provisions of the penal code. [1]

Austria: Cyberbullying is prohibited in the penal code, and the law has been further enacted to safeguard victims and punish online websites that fail to act against offending content. [1]

Romania: Domestic violence legislation within the states expressly defines cyberbullying, while additional provisions to the criminal law for stalking, blackmailing, and threats exist. [1]

Slovakia: Penal law provides for criminalization of cyberbullying, sanctioning based upon the severity of harassment and on the effect produced by the inflection of damage to victims. [1]

1.5.3 Efforts Across the EU

The rest of the EU countries have made legislative and non-legislative efforts to combat cyberbullying:

Greece and Cyprus: Adopted new legislation regarding bullying, which includes cyber bullying. [1]

Finland and Denmark: Prevention and online training programs as a priority. [1]

Netherlands and Belgium: Apply a mix of school security laws, public campaigns, and judicial remedies to deal with online bullying. [1]

Spain, Sweden, and Portugal: Criminal codes cover online harassment under broader categories such as stalking, defamation, and unauthorized disclosure of information. [1]

1.6 Previous Solutions to Combat Cyberbullying

A common question in the realm of preventing cyberbullying is: What works to stop it? Based on more than 15 years of research, a number of evidence-based insights have been found. But it's also vital to think about how young people see things, as they are the ones who are most directly affected by cyberbullying. To come up with useful, youth-informed solutions, it's important to know what students think works and, even more importantly, what has worked for them personally.

A recent national study asked 5,700 middle and high school students from all throughout the United States to give their thoughts. Out of the 1,880 people who said they had been cyberbullied at some point in their lives, about 1,300 students (almost one-third of the total sample) gave extensive, qualitative answers about the activities or techniques that helped reduce or stop the harmful behaviors.

A full review and thematic analysis of these responses put them into a number of different groups. The chart below shows the strategies that are most often mentioned. These are useful since they show what students think are the best ways to deal with cyberbullying.[5]

Block

Blocking the person who was cyberbullying was the most common way that teens said they dealt with it. About one-third of the people who were bullied online said that this method helped make things better. A 16-year-old from Ohio said, "Blocking the person let me end it." After that, they made a new account and said horrible things. They stopped after I said I would tell people on Twitter. They must have stopped because they were scared of getting kicked off Twitter. A 17-year-old from New Jersey said, "Based on my experience, the best thing to do was to block that person and get rid of all the evidence of the bullying (comments, messages, etc.)." After that, I talked to a close friend about how and why it affected me in a kind way before moving on. These stories show not only how effective blocking is seen as a response, but also how important it is to get emotional support after these kinds of events. Blocking people you know in real life who are cyberbullying you may be hard, but it is still a good idea. In these situations, it's also a good idea to keep proof of the bullying and tell both parents and school officials about it.

Ignore

Ignoring the cyberbullying was another common way that teens said they dealt with it. Many of the people who took part said that the harassment stopped when they didn't answer. A 16-year-old girl from Texas said, "After a while, they left me alone." They must have been bored when they didn't get the answer they wanted. A 12-year-old girl from Texas said the same thing: "I pretended it didn't bother me and ignored it." If they can't get a response from you, they'll leave you alone and move on to someone else.

A 15-year-old kid from Virginia agreed, saying, "One of the best ways to get it to stop is to ignore it, but you should always tell someone (parents or friends) so that someone else knows what might be going on." This point of view supports the idea that bullies are often motivated by the reactions they get and therefore not showing emotions can be a good way to stop them.

A 17-year-old girl from Illinois added to this tactic by saying, "Blocking or avoiding the bully at first is a good idea." You should also try to ignore their bullying by staying calm and not getting into fights. If this doesn't work, I tell a friend or my mom, who will help me deal with the bully and try to handle the issue in a friendly and private way. If this doesn't work, I go to Plan 3 and tell my counselor, who will use professional resources to help. Depending on the situation and what the bully does, I switch between the easiest way to deal with it and a more powerful way to deal with it. It's great that you're able to stay calm, but you need to be determined.

These stories show a flexible, phased way to deal with cyberbullying, starting with not getting involved and going up to getting help from trusted people or professionals when needed.

Get Parents Involved

About half of the students who said they had been bullied online said they had told one or both of their parents. This is the highest rate of parents who answered the poll in all of its versions. Also, getting parents involved seemed to work in a lot of cases. Almost 17% of people who answered said that telling their parents was a helpful way to deal with cyberbullying.

A 13-year-old boy from Louisiana said that telling his parents about the problem was the most crucial thing he did. He said, "My feelings were just getting worse when I tried to ignore or suppress it." A 13-year-old from California said, "My parents asked me how I felt about it and

[5] https://cyberbullying.org/teens-talk-works-stop-cyberbullying

what we should do when I told them what was going on." I finally told Xbox Live about this person and blocked them. A 12-year-old boy from Colorado told about how his parents helped him see things in a different way: "My mom and dad just talked to me about how people who do that kind of thing have bigger problems and are just trying to hurt you to feel better." Because of it, I could see through the bad things that other people say and do.

In some cases, parents took the initiative by talking to the families of the people who did the crime. A 12-year-old boy from Arkansas said, "My dad showed me how to block the person on Facebook and on my phone." He then called the person's parents and said he would call the police if it happened again. A 13-year-old girl from New York said the same thing: "It stopped when my mom told the kid's parents about it." She stated she would call the police and show them the proof of the bullying if it didn't stop. She had preserved everything as proof.

There is considerable disagreement about whether getting the bully's parents involved is a good idea, but these stories show that it can be helpful in some situations. A girl from Illinois who was 16 years old said, "My parents talked to the other student's parents." My parents told me to block the student and change my screen name and passwords. I think it was great that my parents called the other student's parents and went right to the source.

These testimonies show how important it is for parents to be involved, not just to give emotional support but also to take practical action to help fix the problem.

Take a Tech Break

Many teens said they stopped using digital platforms for a while or even deleted their accounts because of cyberbullying. A 13-year-old boy from Georgia said, "I blocked the other player and didn't play Xbox Live for a few days." In the same way, a 16-year-old lad from California said, "I would log off for a while and change my email or username. That seemed to work best."

People are usually careful about telling kids to stay away from technology when they are being bullied online because digital platforms are such an important part of social, academic, and recreational life. However, these stories show that taking a break from technology can sometimes help and keep kids safe. It is crucial to highlight, though, that these kinds of actions may end up punishing the victim instead of the person who did the crime.

A 14-year-old girl from Indiana said the same thing: "My mom made me stop using all social media, which wasn't fair to me as punishment for the kids being mean to me, but she said that was the only way to avoid it because the kids' parents weren't taking it seriously and neither were the police." Her story shows how systemic problems can happen when parents, government, or institutions don't do a good job of dealing with cyberbullying.

These points of view make it clear that we need solutions that are more sophisticated and focus on protecting victims while also making sure that young people aren't unfairly kept out of digital places.

Report

Another popular tactic was reporting abusive activity on the app, game, or website. Students admitted that interacting in these virtual spaces was a significant aspect of their life, and some were put off by the possibility of losing their profile or being removed from a game. "Blocking the person and reporting them for abuse to the site usually works," a 16-year-old lad from New York remarked.

Since social networking is their life, blocking, muting, or banning them will likely resolve the issue online but may also result in more serious issues in real life. In a similar vein, a 16-year-old Kentucky kid stated: "When someone starts saying offensive things in game chat, reporting them usually gets them banned instantly and the problem is over." While several noted that it is frequently simple to establish a new online identity that can be used to carry out the cruelty, this didn't seem to happen very frequently.

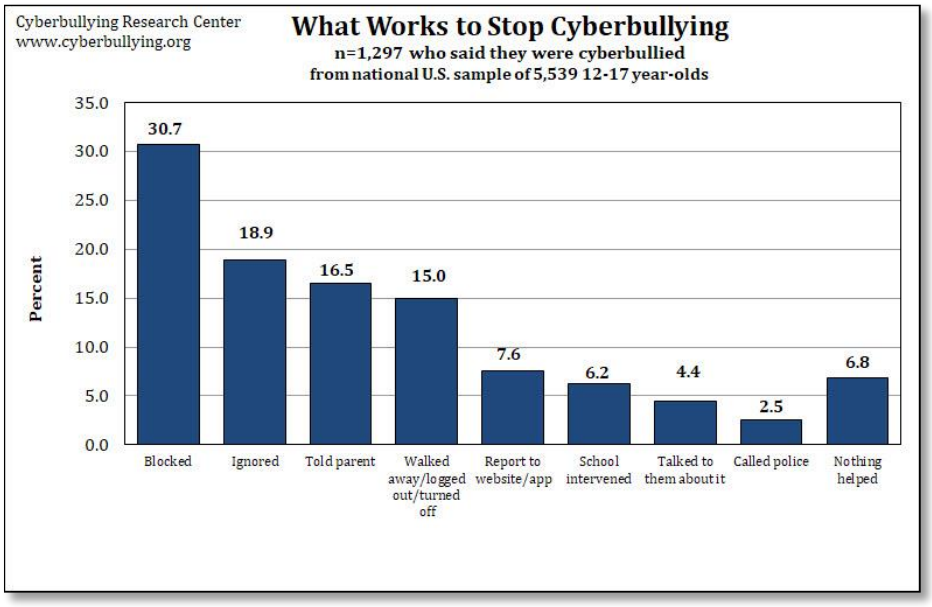


Figure 1.3 What Works to Stop Cyberbullying [6]

1.7 Future Strategies to Combat Cyberbullying

Teenagers can use a number of different methods to deal with and stop cyberbullying. Here are a few examples:[7]

Educate yourself first: Understanding the precise definition of cyberbullying is essential to prevent its occurrence. Investigate the locations and circumstances in which cyberbullying is most prevalent, along with its definition.

ENSURE YOUR PASSWORD IS SAFE. It is essential to safeguard personal and confidential information, such as passwords, to reduce the risk of cyberbullying and other forms of online harm. Sensitive data should never be left visible or accessible to others, and it should not be shared—even with close friends. If there is any indication that such information may have been compromised, it is crucial to update passwords immediately to maintain account security and personal privacy.

USE "PG" PHOTOS. Think about whether you would want your parents, grandparents, and the entire world to see that sultry photo of you before emailing or uploading it. This image can be used as fuel by those who wish to harm you to make your life miserable.

DO not open unsolicited or unidentified messages. Never respond to emails, texts, Facebook messages, or other correspondence from strangers or from persons you know who frequently harass others. Don't read them and delete them. If you open them, they can include viruses that infect your device automatically. Additionally, you should never click on links to websites supplied by strangers. These might also include a malware that installs a backdoor on your device or gathers your private or personal data.

END YOUR ACCOUNT ONLINE. For convenience, avoid saving passwords in form fields on websites or in your browser. Additionally, don't leave your computer or phone logged in when you leave. Don't even allow someone to use your smartphone to pretend to be you online. If you use the library computer and neglect to log out of Twitter, the next individual utilizing the computer could access your account and cause serious issues for you.

---

[6] https://cyberbullying.org/teens-talk-works-stop-cyberbullying
[7] https://cyberbullying.org/Top-Ten-Tips-Teens-Prevention.pdf

TAKE A BREAK BEFORE POSTING. Posting anything that could damage your reputation is not advised. People will form opinions about you based on your online persona. On the basis of this, they will also grant or reject you possibilities (jobs, scholarships, internships). **IMPROVE AWARENESS**. To raise awareness about cyberbullying, launch a campaign, form a club, organize an event, or start a movement. Even though you may know what it is, we won't be able to stop it from happening unless others do as well.
IMPROVE YOUR PRIVACY. Limit your internet profile to friends you can trust. You can share certain information with friends only on the majority of social networking sites, such as Facebook and Instagram, but these settings need to be set up to provide the highest level of security. Additionally, don't accept friend requests from random people on Snapchat. **DO YOUR OWN "GOOGLE."** Search for your name frequently on all of the main search engines (e.g., Google, Bing, Yahoo). Take steps to have any content or personal information that could be used to target you taken down before it becomes an issue.

AVOID CYBERBULLYING OTHER PEOPLE. You should treat others the way you want to be treated. You contribute to the issue when you treat people badly online. Don't go to that level; you're better than that. Instead, set greater goals and serve as a role model for others. Both now and in the future, it is extremely important.

1.8 Case Studies and Real-Life Examples of Cyberbullying

We can be certain that many instances of cyberbullying go undetected. These are the worst instances of cyberbullying that authorities have documented.

Meier, Megan (1992–2006) Megan Taylor Meier, an American teen, hanged herself three weeks prior to her fourteenth birthday. After her parents insisted on an investigation, it was found that her cyberbullying originated on the social networking site MySpace. This instance served as a precursor to the negative consequences of social media.

Prince Phoebe, 1994–2010 The Massachusetts state senate enacted stricter anti-bullying laws and punished six teenagers with civil rights violations after Phoebe Nora Mary Prince committed suicide on January 14, 2010.

From 1996 to 2012, Amanda Todd Due to persistent abuse and blackmail from online users, 15-year-old Canadian student Amanda Michelle Todd killed herself after becoming the victim of cyberbullying.

Parsons, Rehtaeh (1995–2013) On April 4, 2013, 17-year-old Rehtaeh Parsons, a former student at Cole Harbour District High School, attempted suicide by hanging herself in her Dartmouth, Nova Scotia, Canada, home. Targeted online harassment is the cause.

Clementi, Tyler (1991–2010) Tyler Clementi, an American undergraduate at Rutgers University-New Brunswick, committed suicide on September 22, 2010, by plunging from the George Washington Bridge across the Hudson River. Ravi and Wei, Clementi's doom room pals, set up a webcam in his room and harassed him online.

Rodemeyer, Jamey (1997–2011) Prior to his death, Jamey Rodemeyer attended Heim Middle School and was a student at Williamsville North High School. Because he was open about being gay, he endured horrible bullying.

From 1994 to 2019, Sulli When police found Choi Jin-Ri, better known as Sulli of the K-pop girl group f(x), dead in her Seongnam home, they began to look more closely at her career as a feminist and singer who was regularly the target of online cyberbullying and violent Internet trolls.

Unsworth, Tyrone (2003–2016) Tyrone Unsworth, an eighth-grader in Brisbane, Australia, killed himself on November 22, 2016, after years of homophobic harassment. His grandfather had planned for him to attend school on this specific day, but he chose to remain at the farm instead. Around 1:00 p.m., his grandfather found Unsworth dead when he got home from work.

Kimura Hana (1997-2020) At the age of 22, one of six cast members of the Netflix reality series Terrace House, which once aired on FujiTV and featured an oddball bunch of strangers living together, passed away. Kimura received a lot of criticism for her pink hair, toned body, and vivacious personality, which made her a well-known public figure in Japan. Her passing brought attention to the demands placed on women to live up to social norms and cyberbullying.

Fraser, Kelly (1993-2019) Kelly Fraser, a 26-year-old Canadian Inuk pop singer from Igloolik, Nunavut, gained notoriety for both translating popular songs into Inuktitut and performing original songs. Fraser was found dead at her home in Winnipeg, Manitoba. Fraser's death, which was later determined to be a suicide, was attributed by her family to "childhood traumas, racism, and ongoing cyberbullying."

1.9 Modern Challenges in Cyberbullying Prevention

Cyberbullying is becoming a significant public health concern in adolescents, with various researches pointing to its adverse effect on mental health, including increased depressive mood, anxiety, loneliness, and suicidal ideation. Both prevention and intervention of cyberbullying have been established, for example, through education workshops and the utilization of technology in filtering and blocking harmful internet connections. But most such interventions fail to adequately involve youths and grapple with the dynamics of cyberbullying.

One of the solutions can be utilizing artificial intelligence and machine learning. Instagram, for example, uses features that prompt a user to send warnings to her or him prior to posting probable cyberbullying comments and take a break for a while. And along similar lines, the ReThink app uses AI to identify and detect offending words and nudge users to rethink posting. These are intended to reduce the incidents of cyberbullying by providing instant feedback and encouraging smarter online activity. Intervention is also introduced educationally. According to research, the improvement of adolescents' empathy, self-esteem, and healthy parent-child relations can spark cyberbullying to decrease. However, normal class periods will be considered as inactive and idle by the students, therefore making prevention programs that are more engaging and interactive more necessary.

Although this is so, longitudinal research needs to continue to identify long-term effects of cyberbullying and create effective preventive interventions. The multi-faceted nature of cyberbullying, in terms of multiple relational and environmental aspects, must be tackled with sensitivity to incorporate the adolescents' perceptions as well as teachers' opinions.

Finally, while pedagogy and technology are trying their best to combat cyberbullying, there are required new-generation solutions that carry high appeal for adolescents and harness the multi-dimensional nature of the phenomenon.

1.10 Computer Science in Fighting Cyberbullying

Computer scientists have employed natural language processing (NLP) and machine learning to combat cyberbullying. Jun-Ming Xu, Kwang-Sung Jun, Xiaojin Zhu, and Amy Bellmore have studied the social media data for bullying behavior analysis. PhD research by Junming Sui under the supervision of Xiaojin Zhu was focused on building machine learning methods to analyze and prevent bullying. In addition, Felice Resnik et al. examined social media discussions regarding bullying, even those of celebrities. These advancements provide promising tools for cyberbullying detection and prevention and can be utilized to protect individuals from online victimization. [2]

1.11 Conclusion

Cyberbullying is a new digital-era issue, which occurs on social media, text messages, and online discussion boards. Cyberbullying involves harassment, impersonation, and other forms of hostility that take significant psychological and social effects on the victims. The chapter provides the background of cyberbullying, its cause, and its adverse effects on individuals, particularly students and adults. It highlights how cyberbullying can lead to anxiety, depression, poor performance, and even suicidal tendencies. The chapter also emphasizes the importance of fighting cyberbullying through legal means, social awareness, and technological interventions. Different nations have followed different laws and regulations to fight virtual harassment, though enforcement and adapting to new technologies have been problematic. It is essential to comprehend the underlying causes, forms, and impacts of cyberbullying to formulate effective prevention measures. Potential solutions can include AI-powered content moderation, more robust privacy safeguards, and education programs to educate and encourage responsible internet behavior.

Chapter 2: Machine/Deep Learning

2.1 Introduction

The phrases "deep learning" and "machine learning" are often used interchangeably in the context of artificial intelligence. But that isn't accurate. Therefore, learning the concepts and their distinctions should be the first step for anyone hoping to have a deeper grasp of the topic of artificial intelligence. The good news is that it's not as hard as some articles about the subject make it seem.

2.2 Definition of Machine learning

The broad name for the process by which computers learn from data is machine learning. It explains the nexus between statistics and computer science, where algorithms are utilized to carry out a certain task without explicit programming; instead, they identify patterns in the data and generate predictions when fresh data becomes available. Something as basic as linear regression can serve as a standard machine learning algorithm. Consider, for example, that you wish to forecast your salary based on the number of years you have completed higher education. You must first define a function, such as income = y + x * years of education. Next, provide a set of training data to your algorithm. Data about some people's years of higher education and the corresponding income might be included in this straightforward table. Let your algorithm make the final decision, for example, by using an ordinary least squares (OLS) regression. You may now provide the system with test data, such as your individual years of college, and allow it to forecast your income.[8]
Even while this example seems straightforward, it qualifies as machine learning, and certainly, regular statistics are the foundation of machine learning. Without explicit programming, the system learned to generate predictions based solely on inference and patterns.

In summary, there is a lot to say about machine learning in general:
• At the nexus of statistics and computer science, machine learning gives computers the capacity to learn without explicit programming.
• Supervised and unsupervised learning are the two main types of machine learning challenges.
• An OLS regression is a basic example of a machine learning method.

2.3 Machine Learning Fundamentals

As you can see, there are numerous real-world applications for machine learning because of the various kinds of machine learning approaches. Nonetheless, machine learning's foundations are consistent across methods and applications. In order to create efficient models that perform well when applied to fresh, untested data, these machine learning fundamentals contain crucial elements including data, algorithms, training, testing, and assessment methodologies.

[8] https://levity.ai/blog/difference-machine-learning-deep-learning

2.3.1 Types of Machine Learning

- **Supervised Learning**

Labeled data—that is, input data that has been marked with the appropriate output—is used to train supervised learning algorithms. In order to anticipate the output for fresh data, these algorithms aim to learn a mapping from inputs to outputs. Typical algorithms for supervised learning include[9]:

- One method for forecasting continuous outcomes is linear regression. It fits a linear equation to observed data in order to model the connection between a dependent variable and one or more independent variables.
- For binary classification problems, such as forecasting yes/no outcomes, logistic regression is utilized. It makes use of a logistic function to estimate probabilities.
- Decision Trees: By learning basic decision rules derived from the properties of the data, these models forecast the value of a target variable.
- Random Forests: An ensemble of decision trees that improves model accuracy and overfitting control; usually used for regression and classification.
- Support vector machines, or SVMs, are useful in high-dimensional spaces and are mostly employed for classification, however they may also be applied to regression.
- Neural networks are strong models capable of capturing intricate non-linear interactions. Applications for deep learning make extensive use of them.

- **Unsupervised Learning**

Algorithms for unsupervised learning are used to data sets that lack labeled answers. Determining the inherent structure of a collection of data points is the aim here. Typical methods for unsupervised learning include:

- Clustering: A collection of objects is grouped using algorithms such as K-means, hierarchical clustering, and DBSCAN so that items in the same group are more similar to one another than to those in other groups.
- Association: These algorithms, like market basket analysis, identify patterns that characterize significant amounts of your data.
- Principal Component Analysis (PCA) is a statistical technique that transforms a set of observations of potentially correlated variables into a set of values of linearly uncorrelated variables using an orthogonal transformation.
- A particular kind of neural network called an autoencoder is used to efficiently learn how to code unlabeled data.

- **Reinforcement Learning**

Algorithms that use reinforcement learning learn to make a series of choices. In a potentially complicated and unpredictable environment, the algorithm learns to accomplish a goal. In reinforcement learning, an agent chooses what to do by adhering to a policy, and it gains knowledge from the results of these choices by receiving rewards or punishments.

- The value of an action in a certain state is learned via the model-free reinforcement learning technique known as Q-learning

[9] https://www.simplilearn.com/10-algorithms-machine-learning-engineers-need-to-know-article

- Deep Q-Networks (DQN): This method learns effective policies directly from high-dimensional sensory inputs by fusing Q-learning with deep neural networks.
- Policy Gradient approaches: Rather than evaluating the value of actions, these approaches explicitly optimize a policy's parameters.
- Monte Carlo Tree Search (MCTS): Often employed in games like Go, MCTS is a decision-making technique that uses scenario-playing to determine the best options.

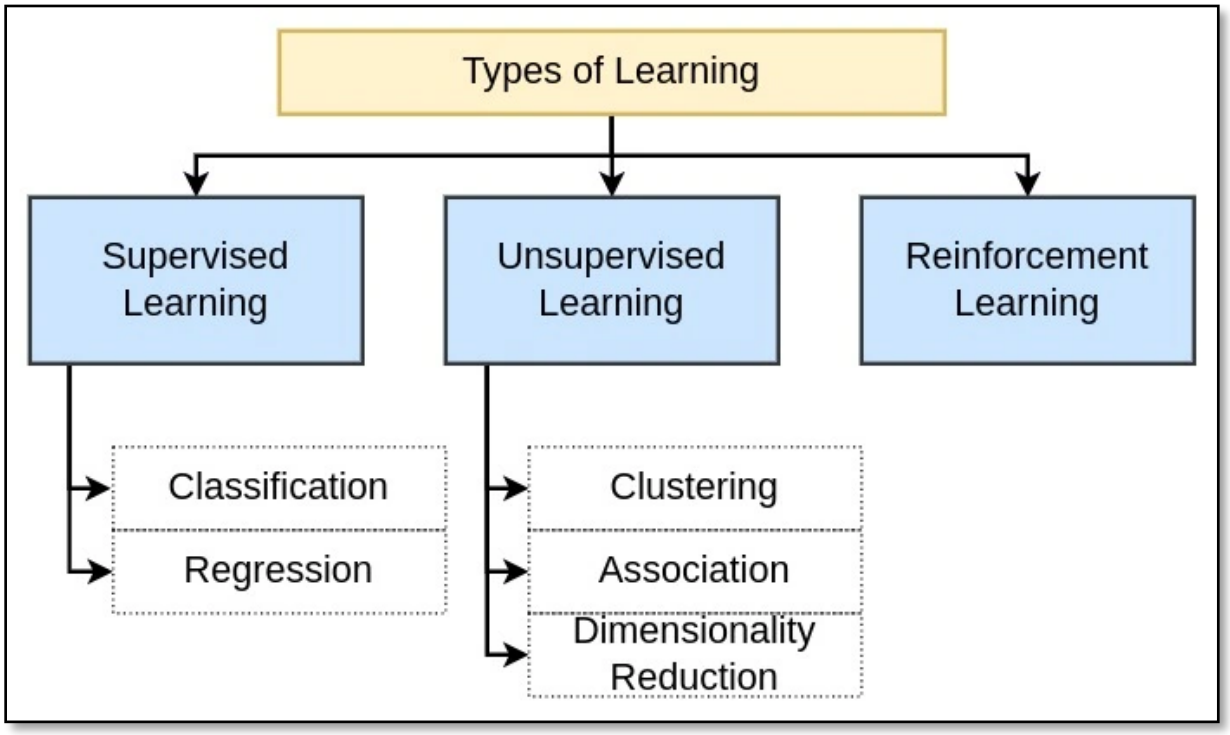


Figure 2.1 Types of Machine Learning[10]

2.3.2 Machine Learning Algorithms :

Below is the list of the top 10 commonly used Machine Learning Algorithms:

- **Linear regression :** Linear regression analysis is employed to forecast the value of one variable depending on the value of another variable.[11]
- **Logistic regression :** Logistic regression calculates the likelihood of an event transpiring, such as voting or abstaining from voting, based on a specified dataset of independent variables.[12]

[10] https://idiotdeveloper.com/types-of-machine-learning/
[11] https://www.ibm.com/think/topics/linear-regression#:~:text=Linear%20regression%20analysis%20is%20used,is%20called%20the%20independent%20variable.
[12] https://www.ibm.com/think/topics/logistic-regression

- **Decision tree :** A decision tree is a straightforward diagram that illustrates several choices and their potential outcomes, facilitating the decision-making process.[13]
- **SVM algorithm :** Support Vector Machine (SVM) is a supervised machine learning technique employed for classification and regression tasks.[14]
- **Naive Bayes algorithm :** Naive Bayes classifiers are supervised machine learning algorithms employed for classification problems, utilizing Bayes' Theorem to determine probabilities.[15]
- **KNN algorithm :** K-Nearest Neighbors (KNN) is a simple but effective classification method that functions by calculating proximity between data points in a feature space. For example, an online streaming company could utilize KNN to forecast whether a new subscriber will cancel their services, depending on demographic data like age.[16]
- **K-means :** K-Means Clustering is a form of Unsupervised Machine Learning algorithm that classifies an unlabeled dataset into unique clusters.[17]
- **Random forest algorithm :** A Random Forest is a collection of decision trees that collectively collaborate to make predictions.[18]
- **Dimensionality reduction algorithms :** Dimensionality reduction enables the reduction of features with the preservation of important information.[19]
- **Gradient boosting algorithm :** Gradient Boosting is an ensemble learning technique for classification and regression. It is a boosting technique that aggregates several weak learners to produce a powerful predictive model.[20]
- **AdaBoosting algorithm :** AdaBoost, which stands for Adaptive Boosting, is a powerful ensemble learning algorithm that aggregates several weak classifiers to create a strong classifier.[21]

[13] https://www.geeksforgeeks.org/decision-tree/
[14] https://www.geeksforgeeks.org/support-vector-machine-algorithm/
[15] https://www.geeksforgeeks.org/naive-bayes-classifiers/
[16] https://www.geeksforgeeks.org/k-nearest-neighbours/
[17] https://www.geeksforgeeks.org/k-means-clustering-introduction/
[18] https://www.geeksforgeeks.org/random-forest-algorithm-in-machine-learning/
[19] https://www.geeksforgeeks.org/dimensionality-reduction/
[20] https://www.geeksforgeeks.org/ml-gradient-boosting/
[21] https://www.geeksforgeeks.org/implementing-the-adaboost-algorithm-from-scratch/

2.3.3 Performance Evaluation Metrics :

When assessing the efficacy and precision of machine learning models, performance measures are essential. They assist gauge a model's performance on a range of tasks and offer insights into its predictive power. The importance of performance metrics, when to use them, and common metrics like precision, recall, accuracy, and F1- score—as well as their formulae and examples—will all be covered in this section.[22]

Understanding Performance Metrics:

Importance:

- Performance metrics quantify the quality of predictions made by machine learning models.
- They provide actionable insights for model evaluation, comparison, and improvement.

Why to use precision and recall instead of accuracy?

An unbalanced dataset cannot be directly replaced by accuracy. For this reason, in order to accurately quantify the model's performance, we must employ precision and recall.

Let's use an example to better grasp this. Our model's efficiency would be 90% if we instructed it to predict class 1 every time out of a dataset of 1000 records, of which 900 belong to class 1 and 100 to class 2.

Applications:

- Many machine learning tasks, such as recommendation systems, clustering, regression, and classification, require performance measures.
- They provide the evaluation of model performance in a variety of fields, including natural language processing, marketing, finance, and healthcare.

[22] https://medium.com/@abhishekjainindore24/a-comprehensive-guide-to-performance-metrics-in-machine-learning-4ae5bd8208ce

Confusion Matrix :

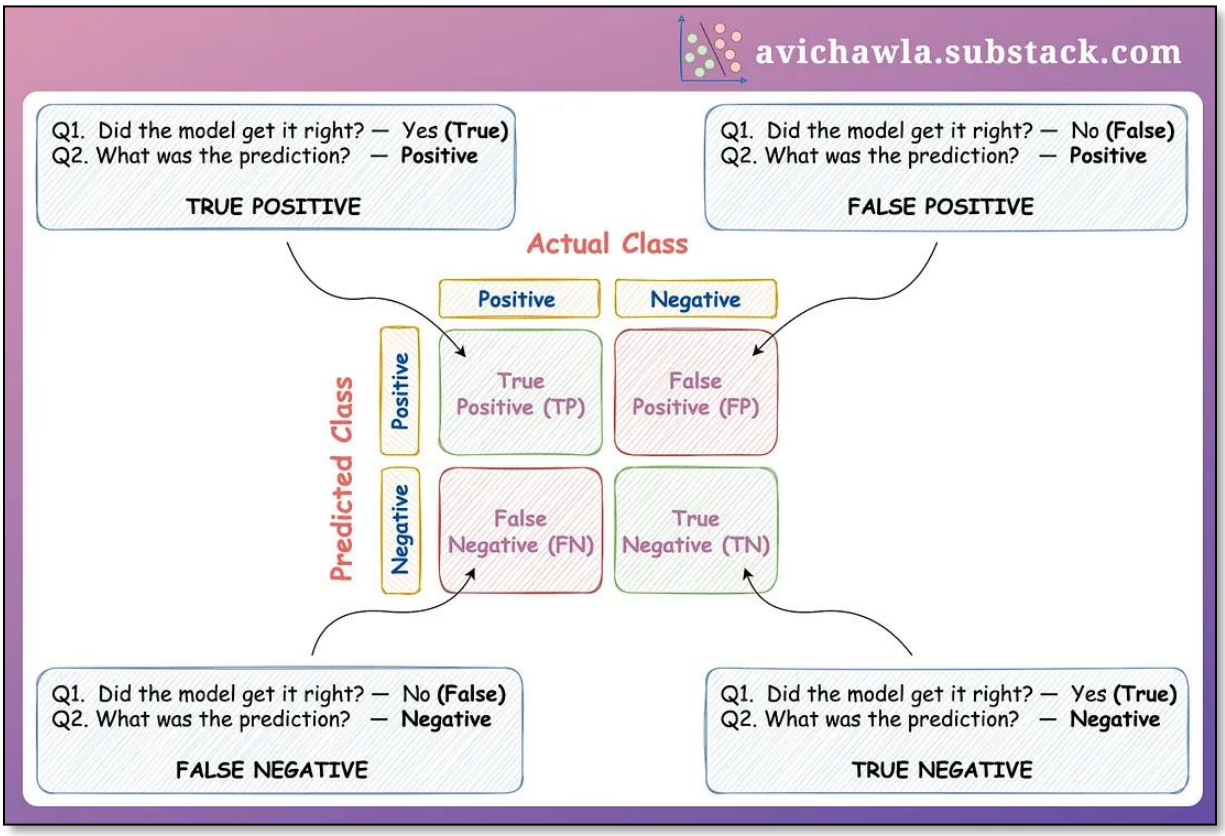


Figure 2.2 Confusion Matrix[22]

Key Performance Metrics:

- **Precision:**

Precision is the percentage of actual positive predictions among all of the model's positive predictions. For instance, accuracy in a spam detection system refers to the proportion of accurately identified spam emails among all emails that are categorized as spam. There are four instances of spam.

Case 1: The email is spam, as predicted by the prediction protocol (TP). In this case, the model's forecast is accurate.

Case2: The email is spam, but the prediction is that it is not (FN). In this case, the user won't experience any problems even if the model's prediction is incorrect.

Case 3: Although the email is not spam, it is predicted to be so (FP). In this case, the user will experience problems because the model's prediction is incorrect. Therefore, the accuracy measure must be utilized in order to reduce FP.

Case 4: The message is not spam, and the prediction is that it is not spam (TN). Precision indicates the proportion of accurately predicted outcomes among all predicted More specifically, we want to lower FP and raise TP and TN.

$$Precision = \frac{TP}{TP + FP}$$

Figure 2.3 Precision [23]

- **Recall (Sensitivity):**

The percentage of accurate positive predictions among all real positive occurrences in the dataset is known as recall. For instance, recall in medical diagnosis refers to the proportion of patients with a particular disease that are correctly diagnosed out of all patients that genuinely have the ailment.

There are four situations where a person may or may not have cancer.

Case 1: A person has cancer, and it is predicted that they will have cancer (TP).

Case 2: The individual is cancer-free and it is predicted that he is cancer-free (TN)

Case 3: Although the individual does not already have cancer, it is predicted that he would develop cancer (FP), which will not result in any problems.

Case 4: Although the individual has cancer, it is predicted that he does not have cancer (FN), which will lead to problems (FN). In order to reduce FN, we must employ recall in this situation.

- How many of the actual results were the ones that were predicted?
- Remember that we want to raise TP and lower FN.

$$Recall = \frac{TP}{TP + FN}$$

Figure 2.4 Recall [23]

- **Accuracy:**

Definition: The percentage of accurate predictions—both true positives and true negatives—among all of the model's predictions is known as accuracy. For instance, accuracy in sentiment

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN}$$

Figure 2.5 Accuracy [23]

[23] https://medium.com/@abhishekjainindore24/a-comprehensive-guide-to-performance-metrics-in-machine-learning-4ae5bd8208ce

analysis refers to the proportion of correctly categorized sentiments (whether positive or negative) among all model predictions.

- **F1- Score:**

Definition: The F1 score is one of the crucial assessment metric utilized in classification issues to measure. It combines precision and recall into a single number.

$$\text{F1 Score} = \frac{TP}{TP + \frac{1}{2}(FP + FN)}$$

Figure 2.6 f1-score[24]

2.4 Definition of Deep Learning

Deep learning algorithms can be considered a theoretically and advanced evolution of machine learning techniques. The field has received much attention in recent years, and for good reason; emerging innovations have led to results that were previously unimaginable. "Deep learning" is a term used to describe algorithms that handle data in a manner conceptually similar to inferential reasoning of humans. It must be mentioned that supervised and unsupervised learning approaches might also yield this outcome. Deep Learning implementations employ an artificial neural network (ANN), a formalized collection of algorithms which are organized in layers, to accomplish this task. The structure of such an ANN is inspired by the biological neural networks of the human brain, making the learning process exhibit a much enhanced capability compared to that of traditional machine learning models.[25]

[24] https://www.v7labs.com/blog/f1-score-guide
[25] https://levity.ai/blog/difference-machine-learning-deep-learning

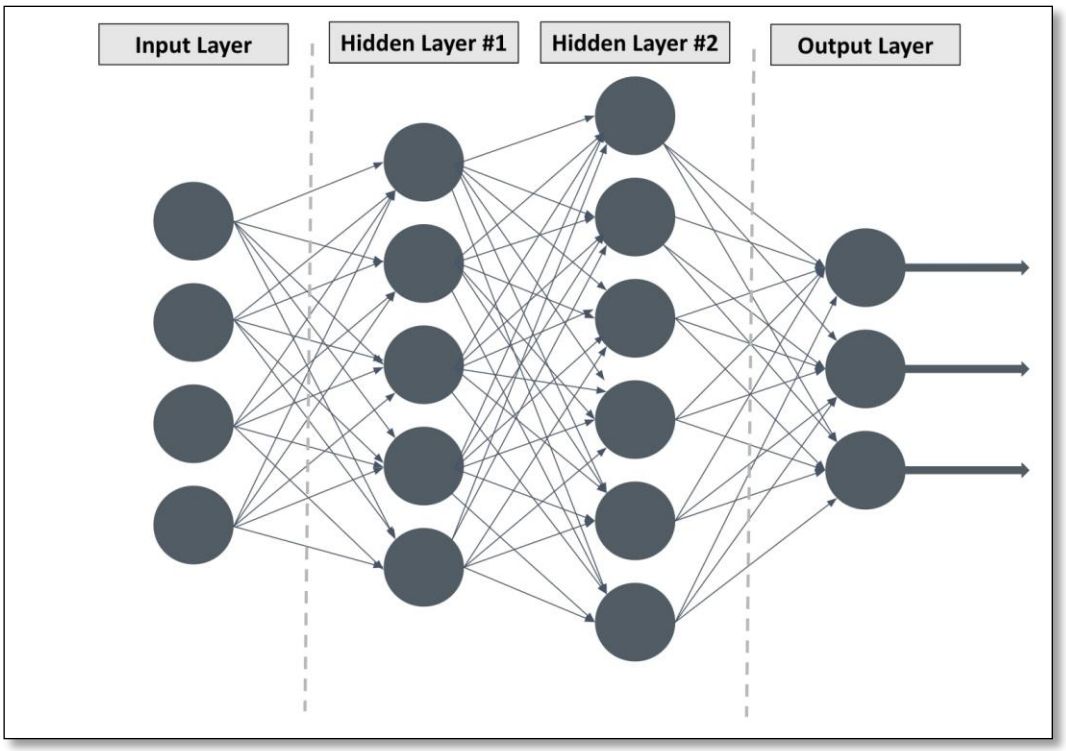


Figure 2.7 A Simple Artificial Neural Network[26]

In the above example, the input layer is on the left and the output layer is on the right. Because the values of the middle layers cannot be observed in the training set, these are what are called hidden layers. Hidden layers are essentially calculated values that the network uses to perform its "magic."
A network has more depth with more hidden layers between the input and output layers. Generally, any ANN with two or more hidden layers is considered a deep neural network.
Deep learning is being applied across a variety of industries today.
For example, deep learning in self-driving cars to recognize pedestrians and STOP signs, deep learning used by the military to identify objects from satellite images, i.e., whether it is safe or not for soldiers to go in, etc. Not to mention, Deep Learning in consumer electronics. Deep Learning techniques, like the ones used by home assistant platforms like Amazon Alexa, are used to understand user interests and accordingly react to voice commands.
Let's consider a more concrete example. Let's say the application of a Deep Learning algorithm by Tesla to allow its cars to recognize stop signs. The ANN would initially identify the relevant STOP sign characteristics, usually referred to as features. Features might be specific elements in the input image, such as edges, points, or objects. An ANN will learn to develop features automatically, whereas a more conventional machine learning algorithm would require a software engineer to select the features that are relevant. Perhaps the initial hidden layer learns to recognize edges, the next one learns to recognize color, and the last one learns to recognize more complicated shapes that are particular to the shape of the object we are attempting to recognize. The Deep Learning algorithms would ultimately learn from their errors, determining the accuracy of the predictions along with the necessity for corrections founded on the training data input.

Deep Learning algorithms, in general, need little human intervention because of their ability to self-learn and engineer features autonomously. Although this reflects deep learning's enormous

[26] https://turnkeystaffing.com/tech-trends/machine-learning-vs-deep-learning/

potential, the two prominent reasons for its recent success are data and processing power availability.
For one, as will be seen in this principle's exceptions, Deep Learning requires truly enormous amounts of data. To take an example, the effective functioning of Tesla's self-driving algorithms requires millions of images and countless hours of video.
Second, Deep Learning needs significant computing power. Advances in cloud computing hardware and the advent of high-performance GPUs (graphic processing units, which enable computations to be faster), however, have the potential to reduce the time needed to train a Deep Learning network from weeks to a few hours.
However, the advent of transfer learning—that is, making use of pre-trained models—has arguably been the most vital development in deep learning. The rationale is that transfer learning can be thought of as a fix to the necessity of huge training datasets, which were required to enable ANNs to yield worthwhile outcomes.
Previously, ANN algorithms were not considered to be the optimal solution to every issue due to these enormous data demands. Nevertheless, pre-trained models can today be utilized to fulfill this data requirement for most applications.

In conclusion:

- A specific subset of machine learning is called deep learning.
- An artificial neural network, a layered framework of algorithms, is the foundation of deep learning.
- Although deep learning takes a lot of data, it can operate with minimal human involvement. The solution to the problem of big training datasets is transfer learning.

2.5 Deep Learning Fundamentals :

Deep Learning is a subset of Machine Learning that use Neural Networks (NNs) to autonomously acquire knowledge from vast datasets. Deep Learning excels in image recognition, natural language processing, and robotics.

2.5.1 Neural Networks and Deep Neural Networks:

What Is Neural Network ? :

Machine learning models called neural networks replicate the intricate workings of the human brain. These models process data, identify patterns, and facilitate activities like pattern recognition and decision-making through the use of interconnected nodes or neurons.[27]

Understanding Neural Network In Deep Learning :

Without the need for pre-established rules, neural networks may directly learn from input and recognize patterns. These networks are constructed using a number of essential elements:

- **Neurons:** The fundamental units that take in information, each neuron is controlled by an activation function and a threshold.
- **Connections:** Information-carrying links between neurons that are controlled by biases and weights.

[27] https://www.geeksforgeeks.org/neural-networks-a-beginners-guide/

- The strength and impact of linkages are determined by weights and biases.
- Mechanisms that assist in processing and transferring information between layers of neurons are known as propagation functions.
- The learning rule is a technique that gradually modifies biases and weights to increase accuracy.

Neural network learning is a three-stage, systematic process:

- **Input Computation:** The network receives data.
- **Output Generation:** The network produces an output based on the parameters that are in place at the moment.
- **Iterative Refinement:** The network continually improves its performance on a variety of tasks by fine-tuning its output by modifying weights and biases.

In an adaptive learning environment:

- A simulated scenario or dataset is presented to the neural network in an adaptive learning environment.
- In reaction to fresh information or circumstances, parameters like weights and biases are modified.
- The network's response changes with each modification, enabling it to successfully adjust to various tasks or circumstances.

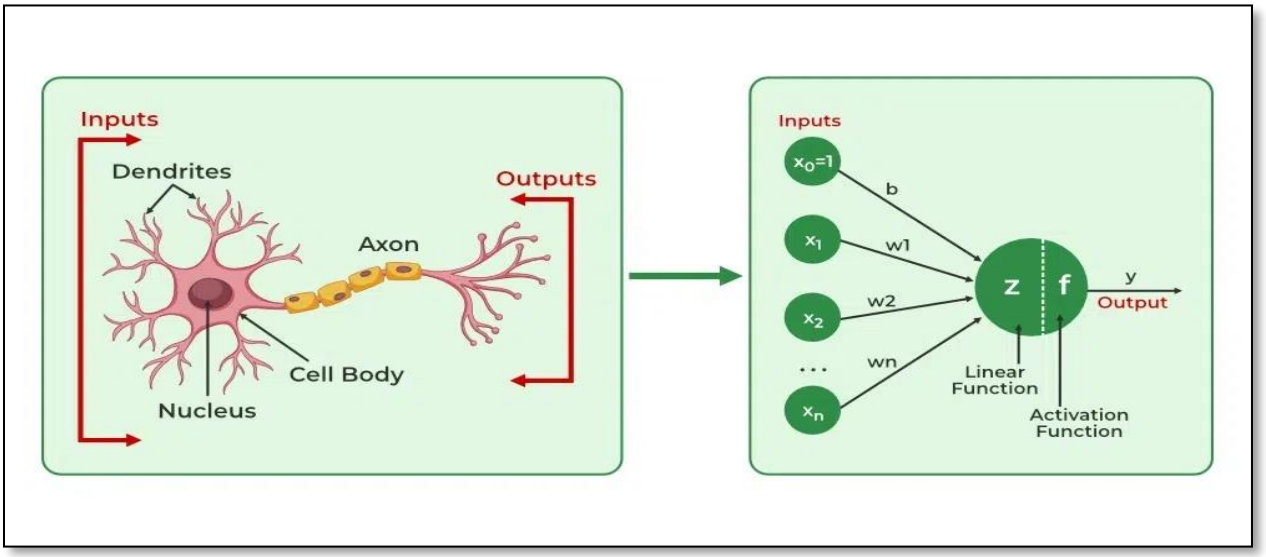


Figure 2.8 the analogy between a biological neuron and an artificial neuron, showing how inputs are received and processed to produce outputs in both systems. [28]

2.5.2 Importance of Neural Networks :

Neural networks are essential for identifying intricate patterns, addressing complicated issues, and adapting to evolving circumstances. Their ability to assimilate vast quantities of data is transformative, impacting automated decision-making, autonomous vehicles, and natural language processing, among other technologies. Neural networks enhance decision-making,

[28] https://www.geeksforgeeks.org/neural-networks-a-beginners-guide/

increase productivity, and optimize processes across diverse sectors. They persist in driving innovation and shaping the trajectory of technology as the cornerstone of artificial intelligence.

2.5.3 Neural Network Architecture Layers :

- The network receives its input data at the input layer. Each input neuron in the layer represents a feature in the input data.
- The predominant computational burden is carried out by the hidden layers. A neural network may contain one or more concealed layers. The neurons comprising each layer transform inputs into a format usable by the output layer.
- Output Layer: The final layer produces the model's output. The outputs vary in format based on the specific objective (e.g., classification, regression).

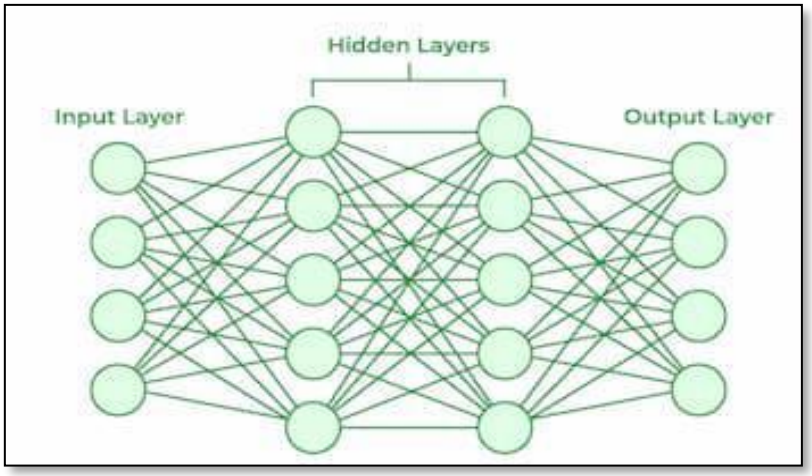


Figure 2.9 Layers in Neural Network Architecture[28]

2.6 The main differences between Machine Learning and Deep Learning

Machine learning encompasses algorithms, including deep learning algorithms. Therefore, it may be advantageous to examine the distinctive characteristics of Deep Learning within the realm of machine learning. The architecture of the ANN algorithm, the increased data demands, and the reduced necessity for human intervention are the responses. Primarily, Deep Learning is founded on an artificial neural network, whereas conventional Machine Learning methods, such as linear regression or decision trees, possess a relatively simple structure. This multi-layered artificial neural network is complex and interrelated, like to the human brain.

Secondly, algorithms employing Deep Learning require significantly less human intervention. A software engineer would manually choose features and a classifier to categorize images, confirm that the output meets requirements, and adjust the algorithm if it does not, assuming the STOP sign image identification employed a traditional machine learning approach. Nonetheless, as it is a deep learning algorithm, features are extracted automatically, and the program learns from its errors.

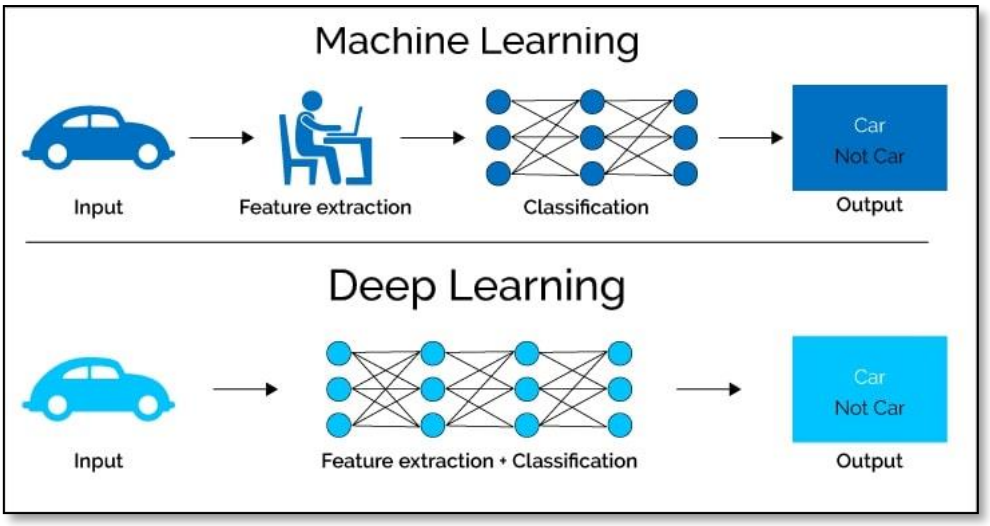


Figure 2.10 Differences between Machine / Deep Learning[29]

Thirdly, Deep Learning requires significantly more data than traditional machine learning algorithms to function effectively. Deep learning often utilizes millions of data points, whilst machine learning often employs only a thousand. A deep learning system necessitates a substantial dataset to mitigate oscillations and yield high-quality interpretations because to its complex multi-layer architecture.

2.7 What Is Natural Language Processing(NLP)

Natural language processing (NLP), a subset of artificial intelligence (AI) and computer science, utilizes machine learning to enable computers to understand and engage with human language. NLP integrates statistical modelling, machine learning, deep learning, and computational linguistics—the rule-based modelling of human language—to enable computers and digital devices to recognize, understand, and generate text and speech. NLP research has facilitated the emergence of generative AI, exemplified by the communication capabilities of large language models (LLMs) and the understanding of requests by picture creation models. A multitude of individuals presently utilize natural language processing (NLP) in their daily activities, evidenced by search engines, voice-activated customer support chatbots, voice-activated GPS systems, and smartphone digital assistants capable of responding to enquiries, such as Microsoft's Cortana, Apple's Siri, and Amazon's Alexa. Moreover, NLP is increasingly vital in enterprise solutions that facilitate the automation and optimization of commercial operations, enhance employee productivity, and simplify organizational processes.[30]

2.8 Explanation of Large Language Models (LLMs) :

Large language models (LLMs), a category of deep learning algorithms, can identify, summarize, translate, predict, and generate content from extensive datasets.

[29] https://medium.com/levity/deep-learning-vs-machine-learning-whats-the-difference-e367803bb96d
[30] https://www.ibm.com/think/topics/natural-language-processing

2.8.1 Definition of Large Language Models (LLM)

Transformer networks constitute a kind of deep learning architectures predominantly exemplified by large language models. A transformer model is a neural network that analyses relationships in sequential input, such as words in a phrase, to comprehend context and meaning.[31]

A transformer comprises many transformer blocks, also known as layers. A transformer has feed-forward, normalization, and self-attention layers that work together to analyze input and predict output streams during inference. Stacking layers enables the creation of more profound transformers and more powerful language models. Google first introduced Transformers in their 2017 paper, "Attention Is All You Need."

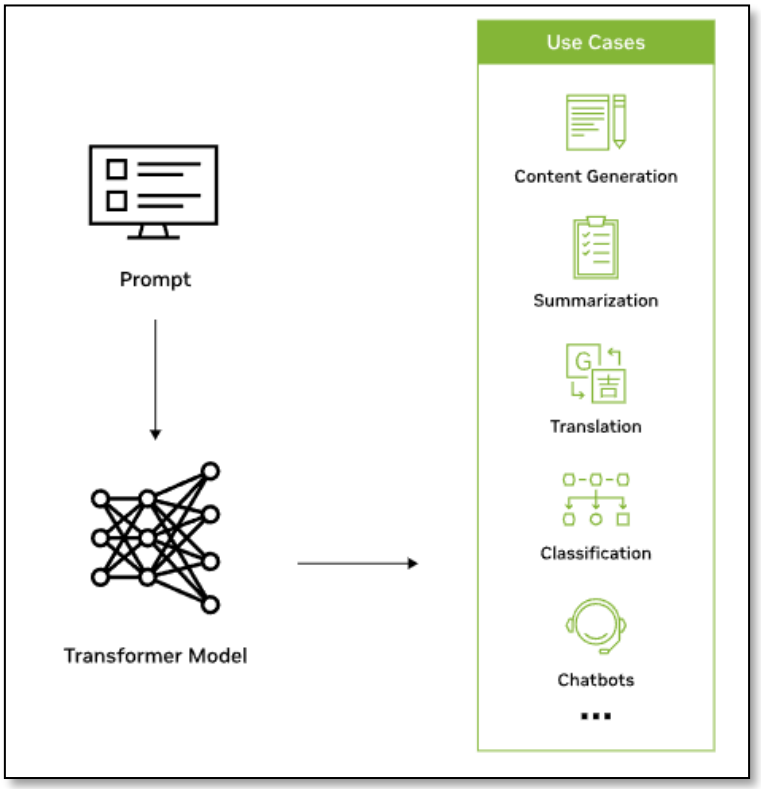


Figure 2.11 How transformer models work. [31]

Transformers excel in managing large language models due to two significant innovations: positional encodings and self-attention. Positional encoding embeds the order of input into a given sequence. Essentially, positional encoding enables the entry of words into the neural network in a non-sequential manner rather than in a sequential order inside a sentence. In the processing phase, self-attention assigns a weight to each element of the input data. This weight signifies the importance of that input in comparison to the other inputs. In other words, models can now focus on the genuinely significant features of the input instead of allocating equal attention to all inputs. As the model analyses and assesses vast quantities of data, it progressively acquires a representation of the elements of the input that the neural network

[31] https://www.nvidia.com/en-us/glossary/large-language-models/

should prioritize. The integration of these two methodologies facilitates the examination of intricate settings and the interrelations among many components across extensive distances in a non-linear fashion. The extensive task can be divided into multiple smaller, simultaneous computations due to the capacity to process data non-sequentially. GPUs are inherently suited for addressing these challenges in parallel, facilitating the processing of extensive transformer networks and unlabeled datasets.

2.8.2 Importance of Large Language Models :

AI models have historically focused on perception and understanding. Recently, the capability of AI models to generate human-like content has been realized using large language models, which are trained on extensive internet-scale datasets comprising hundreds of billions of parameters. To tackle the most formidable global challenges, models possess the ability to read, write, code, design, and create with credibility. They can also augment human creativity and elevate productivity across several industries.

The applications for these LLMs encompass numerous use cases. An AI system can comprehend the language of protein sequences to generate functional compounds that aid researchers in developing novel, life-saving vaccinations. Computers may aid individuals in performing their primary functions: creation, communication, and creativity. A comprehensive language model can assist a writer facing writer's block in generating ideas. A software developer can enhance efficiency by utilizing LLMs to generate code from natural language descriptions.

2.8.3 Large Language Models Examples :

Increasingly complex LLMs have been made possible by developments throughout the computational stack. GPT-3, a 175 billion-parameter model that produced text and code in response to brief written cues, was released by OpenAI in June 2020. With 530 billion parameters, the Megatron-Turing Natural Language Generation 530B, one of the biggest models for reading comprehension and natural language inference in the world, was created by NVIDIA and Microsoft in 2021.[32]

The capabilities of large language models have increased in conjunction with their size. Use cases for LLMs in text-based material can be broadly classified as follows:

- Creation (including narrative development and marketing content production)
- Summarization (including the paraphrase of legal texts or the condensation of meeting minutes)
- Translation (e.g., text-to-code, interlingual)
- categorization (e.g., sentiment analysis, toxicity classification)
- Chatbots (including virtual assistants and open-domain question-and-answer systems)

Global enterprises are increasingly adopting LLMs to provide new opportunities:

- Medical researchers in the healthcare sector build extensive language models for tasks such as protein structure prediction, which can uncover illness patterns and predict

[32] https://www.nvidia.com/en-us/glossary/large-language-models/

outcomes, utilizing a corpus of data derived from textbooks, research publications, and patient electronic health records.

- Retailers may employ dynamic chatbots and large language models to provide customers with exceptional experiences.
- Large Language Models (LLMs) can assist developers in software creation and in directing robots to execute manual tasks.
- Large Language Models (LLMs) can assist financial advisers in drafting transcripts of important meetings and summarizing earnings calls.
- Marketers can instruct a large language model to classify products based on descriptions or to aggregate consumer enquiries and feedback into clusters.

Despite being in their nascent stages, massive language models possess significant potential; a singular model equipped with zero-shot learning capabilities can rapidly understand and generate human-like thoughts, addressing nearly every issue. The potential for value production is immense, as the use cases encompass all businesses, transactions, and industries.

2.8.4 How Do Large Language Models Work

Unsupervised learning is employed for the training of large language models. Unsupervised learning utilizes unlabeled datasets to enable models to discern previously unrecognized patterns in data. A primary challenge in developing AI models is the necessity for extensive data labelling, which is mitigated by this. Due to the extensive training process, LLMs can be utilized for multiple applications without requiring task-specific training. These models are designated as foundation models.

Zero-shot learning denotes the foundational model's ability to generate text for diverse applications with minimum training or instruction. One-shot and few-shot learning are two modalities of this capability wherein the foundational model is provided with one or a limited number of examples demonstrating task execution to enhance its understanding and performance on certain applications.

Despite the remarkable potential of zero-shot learning with large language models, developers and enterprises inherently seek to regulate these systems to function according to their preferences. These extensive language models can be customized through various techniques to provide enhanced precision when utilized for specific applications. The approaches include adapters, fine-tuning, and prompt tuning.

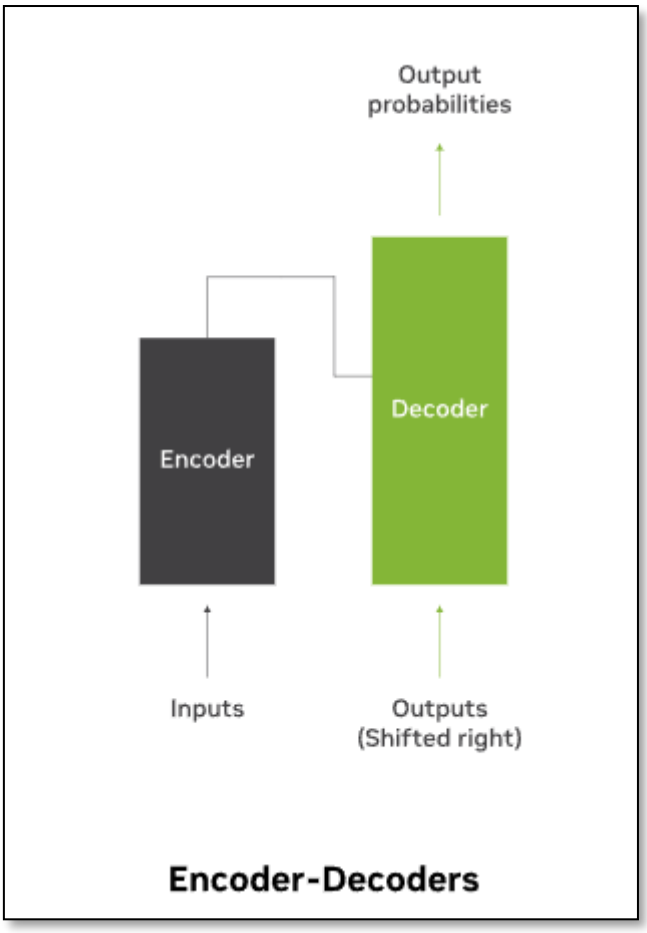


Figure 2.12 Image shows the structure of encoder-decoder language models. [33]

Large language models are categorized into several classes suitable for diverse applications:

- **Encoder-only:** These models are typically suitable for applications such as sentiment analysis and classification that need language understanding. BERT (Bidirectional Encoder Representations from Transformers) exemplifies an encoder-only model.
- **Decoders exclusively:** This category of models specializes in generating content and language. The use cases include the development of blogs and narratives. decoder-only architectures, exemplified by Generative Pretrained Transformer 3 (GPT-3).
- **Encoder-decoder:** These models understand and generate content by integrating the encoder and decoder components of the transformer architecture. This architecture specializes at translation and summarization. The T5 (Text-to-Text Transformer) architecture exemplifies an encoder-decoder model.

2.8.5 The Challenges of Large Language Models:

The majority of enterprises have encountered challenges in market entry due to the significant financial investment, enormous data requirements, technical expertise, and substantial computing infrastructure necessary for the development and management of large language models.

[33] https://www.nvidia.com/en-us/glossary/large-language-models/

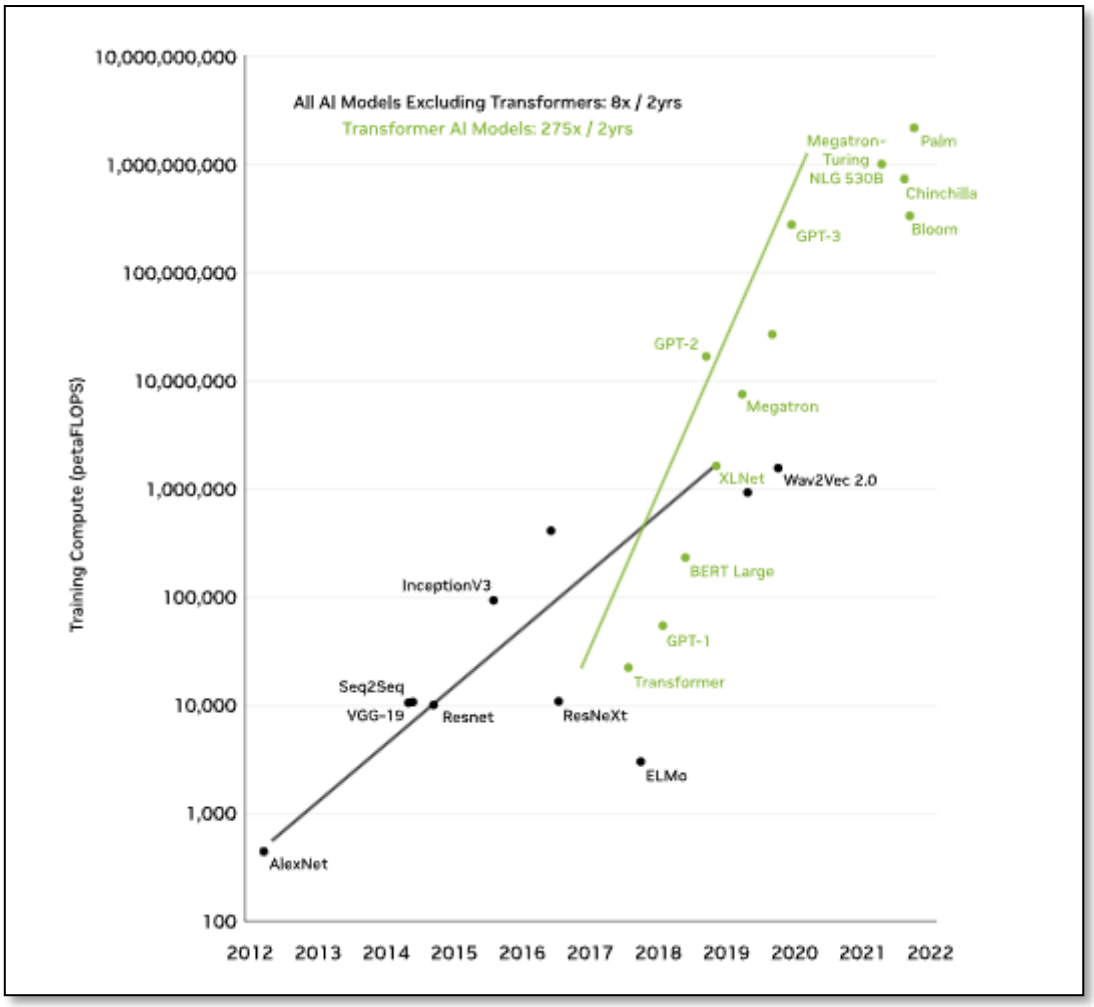


Figure 2.13 Compute required for training transformer models. [34]

- **Workload that is time-, cost-, and computation-intensive:** Maintaining and enhancing LLMs necessitates significant financial resources, technological expertise, and large computational infrastructure. Training a large language model requires weeks to months of concentrated effort and thousands of GPUs. Estimates suggest that a single training run for a GPT-3 model, comprising 175 billion parameters and utilizing 300 billion tokens, might exceed $12 million in computational expenses.
- **Required data scale:** As previously stated, a considerable volume of data is necessary to train a large model. Many organizations encounter difficulties in obtaining sufficiently extensive datasets to train their large language models. This issue is intensified in scenarios involving sensitive data, such as financial or health information. The requisite information for model training may not even exist, in fact.
- **Technical expertise:** Training and deploying large language models is highly complex due to their scale, requiring an in-depth comprehension of deep learning algorithms, transformers, distributed software, and hardware, along with the ability to manage thousands of GPUs simultaneously.

[34] https://www.nvidia.com/en-us/glossary/large-language-models/

2.9 Related Works :

The authors conducted a systematic analysis of 22 research articles on automatic cyberbullying detection to assess the extent to which these studies conformed to the defining characteristics of cyberbullying (e.g., persistence, intentional intent to injure, peer-to-peer). They emphasized considerable limitations: the majority of studies inadequately defined cyberbullying by examining isolated incidents of harassing messages instead of persistent behavior, exhibited methodological deficiencies (e.g., absence of expert annotators or inter-rater reliability), and relied on biassed datasets. Furthermore, grading criteria frequently emphasized overall correctness at the expense of performance specific to cyberbullying. To corroborate these findings, the authors utilized two datasets (Formspring and Bullying Traces) and tested classifiers such as SVM, Logistic Regression, and Random Forests across all feature combinations (textual, sentiment, word embeddings, psycholinguistic characteristics). Results indicated slight enhancements through feature engineering, with F1-scores for cyberbullying detection remaining low (e.g., 0.45 on Formspring), underscoring the ineffectiveness of existing algorithms for practical applications. They advised that future research should focus on datasets that encompass repeated contextual interactions; involve domain experts as annotators; establish standardized evaluation metrics for cyberbullying detection (e.g., CB-class recall/F1-score); and address concerns such as user consent and peer context in data collection. [3]

Paper	Year	Performed Task	Datasets	Dataset's size	Classification model	Annotation Judgment	Features used	Preprocessing steps	Achievements
CyberBERT: BERT for Cyberbullying Identification	2020	Cyberbullying identification (multi-class and binary classification)	Twitter, Wikipedia, Formspring	**Original**: Twitter: 16,090 Wikipedia: 115,864 Formspring: 12,773 **Oversampled**: Twitter: 30,072 Wikipedia: 183,814 Formspring: 18,981	Fine-tuned BERTLarge	Manual annotation	BERT embeddings (token, segment, position)	1. Oversampling (SMOTE) 2. Sentence length truncation (Twitter: 30, Wikipedia: 201, Formspring: 105) 3. Train/Val/Test split (80-10-10)	State-of-the-art F1-scores: Twitter: 0.94 Wikipedia: 0.91 Formspring: 0.92 Outperformed CNN, LSTM, BiLSTM with attention. Reduced computational cost via knowledge distillation.
Multi-Class Imbalance in Text Classification: A Feature Engineering Approach to Detect Cyberbullying in Twitter	2020	Multi-class cyberbullying severity detection (non-cyberbullied, low, medium, high)	Twitter (USA, Texas)	**Original: 2.2M tweets Annotated: 11,904 tweets (low: 404, medium: 590, high: 443, non-cyberbullying: 10,467)**	Naive Bayes (NB), SVM, Decision Tree (DT), **Random Forest (RF)**, KNN **Best Model**: RF with SMOTE + PMI-SO + Predicted Features	Manual annotation by 3 postgraduate students; majority voting for consensus	Network (followers, friends) - Activity (tweet time, account age) - User (predicted age, gender) - Content (profanity) - Personality (Big Five traits) - **PMI-SO** (novel semantic feature)	Lowercase conversion - Reducing repeated letters - Standardizing URLs/hashtags/@mentions - Tokenization (CMU TweetNLP) - SMOTE oversampling (300% minority classes) - Cost-sensitive learning	- **State-of-the-art performance**: 93% accuracy, 92% F1-score, 84% Kappa - Identified key predictors (e.g., age 19–22 for low severity, neuroticism for high severity) - First multi-class approach for cyberbullying severity detection on Twitter
Cyberbullying Detection Based on Emotion	2023	Cyberbullying detection using emotion, sentiment, and contextual features	1. **Cyberbullying Datasets**: - Toxic (Wikipedia) - Twitter (Hate Speech) 2. **Emotion Datasets**: - CBET - TEC	**- Toxic Dataset: - Original: 214k comments (10% CB) - After sampling: 21.8k CB, 65.4k non-CB - Twitter Dataset: - Original: 25k tweets (83% CB) - After sampling: 20k CB, 9k non-CB - Emotion**	BERT (base, large), XLNet, DL models (Keras/TensorFlow)	- **Cyberbullying Datasets**: Existing labels (Kaggle/CrowdFlower). - **Emotion Datasets**: Validated via sentiment analysis (AFINN), emotion lexicon (NRC), and DL model (EmoInt).	- Contextual (BERT embeddings) - Emotion (EDM predictions, NRC lexicon) - Sentiment (AFINN polarity/score)	- **Data Cleaning**: Remove punctuation/symbols, lemmatization, lowercase conversion, digit removal. - **Sampling**: Under-sampling (Toxic) and oversampling (Twitter). - **Emotion Validation**: Three-step process (sentiment, lexicon, DL model).	- Improved recall by 0.5–0.6 and F1-score by 0.7 over SOTA models. - Created emotion-annotated dataset (VED). - Demonstrated effectiveness of emotion features for CB detection.

				Dataset (VED): 32k instances					
Class Imbalance-sensitive Approach Based on PLMs for Cyberbullying Detection	2024	Cyberbullying detection using PLMs and handling class imbalance in English & Arabic	1. **English**: - Formspring - Impermium 2. **Arabic**: - Aljazeera	**- Formspring:** **- 13,156 instances (2,035 CB, 11,121 Non-CB)** **- Impermium:** **- 6,580 instances (1,742 CB, 4,838 Non-CB)** **- Aljazeera:** **- 6,186 instances (533 CB, 5,653 Non-CB)**	- Multi-channel CNN (MCNN) - Bidirectional LSTM (BLSTM) combined with MCNN - Fine-tuned PLMs (BERT, RoBERTa, DistilBERT)	- **Formspring**: Labeled by Amazon Mechanical Turk workers. - **Impermium**: Public Kaggle dataset. - **Aljazeera**: CrowdFlower annotations for offensive/clean classes.	- Fine-tuned PLM embeddings (BERT, RoBERTa, DistilBERT) - Text data augmentation (EDA) - Cost-sensitive learning and Focal Loss	- **Formspring**: Remove "Q:/A:" markers, lowercasing, HTML/special character removal. - **Impermium**: Hyperlink removal, text standardization. - **Aljazeera**: Arabic text normalization (diacritics, symbols, letter normalization).	- Achieved **78%-96% Recall** on unbalanced datasets. - Outperformed SOTA models on Formspring (F1-score: 0.57–0.93). - Introduced effective class imbalance handling via EDA, cost-sensitive learning, and Focal Loss.

RoBERTaNET: Enhanced RoBERTa Transformer Based Model for Cyberbullying Detection With GloVe Features	2024	Cyberbullying detection in tweets	Kaggle Cyberbullying Classification Dataset	**~47,000 tweets (6 classes, 8,000 instances per class)**	RoBERTa (proposed), RF, SVM, KNN, NB, BiLSTM, ConvLSTM, BERT, CNN	Human-annotated (categories: Gender, Age, Religion, Ethnicity, Other, No bullying)	GloVe, Word2Vec, FastText embeddings	NLP preprocessing (unspecified steps)	- **95.9% accuracy** (RoBERTa + Word2Vec) - **95% precision**, **97% recall**, **96% F1-score** - **95.07% mean accuracy** via 5-fold cross-validation - Outperformed state-of-the-art models (e.g., Bi-LSTM, BERT, SOSNet)

Table 2.1 Related Works table

2.10 Conclusion

This chapter highlighted the revolutionary potential of machine learning/deep learning architectures and large language models in preventing cyberbullying. The integration of solid algorithmic designs, responsible AI practices, and ongoing bias reduction innovations provides scalable, proactive solutions for improving online safety while protecting privacy and free expression rights. The technical assumptions presented in this chapter form the foundation for the implementation of LLM-based cyberbullying detection systems proposed in the overarching project.

Chapter 3: Technical details & Proposed Approach

3.1 Introduction

This chapter provides a comprehensive overview of technical architecture, equipment, data sets, and methodologies utilized in developing a machine learning model for identifying instances of cyberbullying in online content. This study seeks to improve the detection and moderation of offensive, abusive, or harmful content that might adversely affect individuals and communities in online groups. The suggested methodology mixes advanced natural language processing (NLP) with various LLMs, enabling the model to analyze and comprehend the complexity of human language across many web platforms. The system utilizes linguistic features, semantic interpretation, and contextual comprehension to differentiate between benign interactions and those filled with cyberbullying terminology. The subsequent sections will delineate data selection and preprocessing, algorithm model selection, assessment metrics, and implementation tools that contribute to the overall accuracy and efficacy of the detection system.

3.2 Methodology adopted

This initial contribution outlines an approach for binary text classification of messages/comments into two categories: cyberbullying (CB) and non-cyberbullying (NCB), utilizing two datasets.

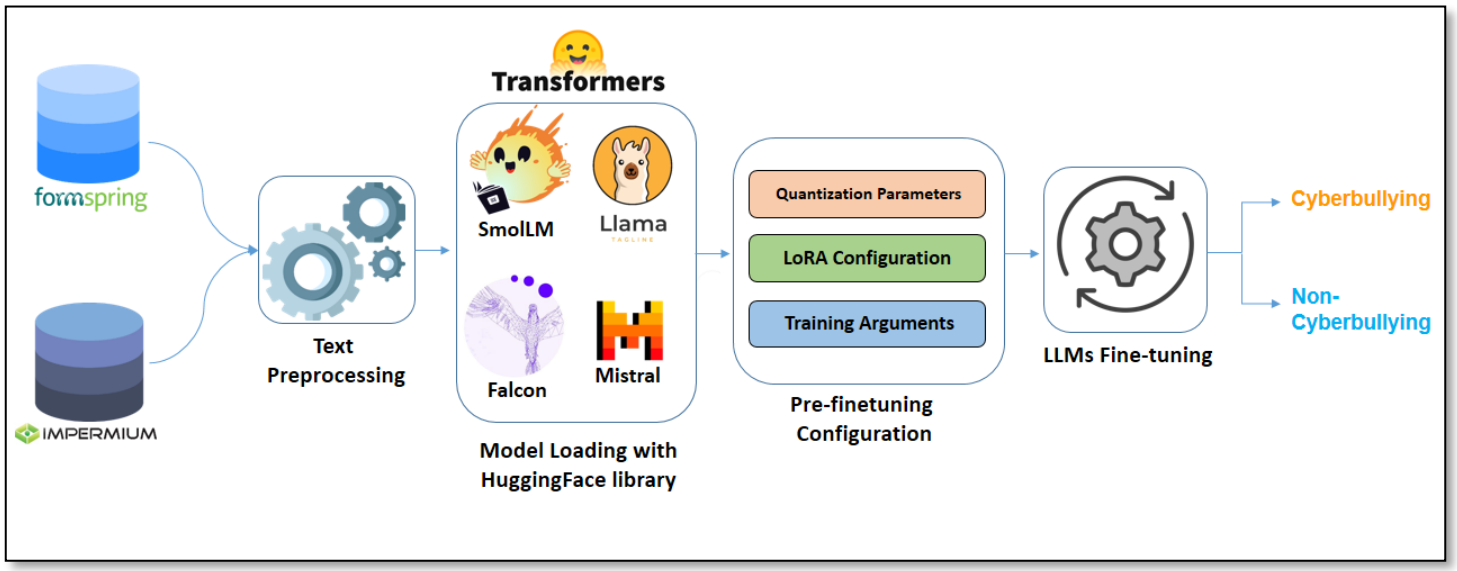


Figure 3.1 Fine-Tuning Large Language Models for Cyberbullying Detection

3.2.1 Used Libraries

Hugging Face : Hugging Face is a community and platform for data science and machine learning (ML) that aids users in developing, deploying, and optimizing machine learning models. It provides the essential framework for showcasing, executing, and deploying artificial intelligence (AI) in practical applications. Furthermore, users can examine previously submitted models and data sets. Hugging Face is often termed the GitHub of machine learning due to its facilitation of unrestricted sharing and testing of developers' work.[35]

[35] https://www.techtarget.com/whatis/definition/Hugging-Face

Transformers : Transformers is a compilation of pretrained models designed for training and inference in computer vision, audio, natural language processing, and multimodal applications. Develop inference applications, generate text utilizing large language models, and train models on your data with Transformers.[36]

Datasets : The Datasets library provides streamlined access to and distribution of datasets across various machine learning domains, including computer vision, audio processing, and natural language processing (NLP). Users can swiftly import datasets and prepare them for deep learning model training using sophisticated, modular data processing pipelines with a single line of code. The library enhances efficiency for extensive data management by employing the Apache Arrow format, facilitating zero-copy reads and accommodating virtually limitless memory capacity. Moreover, seamless interaction with the Hugging Face Hub facilitates the importation and dissemination of datasets throughout the broader machine learning community, thereby promoting collaboration and reproducibility in research.[37]

PEFT : The PEFT (Parameter-Efficient Fine-Tuning) library enables the effective adaptation of large pretrained models for diverse downstream applications without necessitating the fine-tuning of all model parameters, which is prohibitively costly. PEFT methodologies yield performance comparable to fully fine-tuned models while only adjusting a limited number of additional parameters, significantly diminishing computational and storage expenses. Consequently, large language models (LLMs) can now be trained and stored more conveniently on consumer hardware. [38]

BitsandBytes : The bitsandbytes package is a minimalistic Python wrapper for CUDA special functions, specifically 8-bit optimizers, matrix multiplication (LLM.int8()), and 8 & 4-bit quantization methods.

The library features quantization primitives for 8-bit and 4-bit operations via bitsandbytes.nn.Linear8bitLt and bitsandbytes.nn.Linear4bit, as well as 8-bit optimizers through the bitsandbytes.optim module. Efforts are underway to enhance support for additional hardware backends, specifically Intel CPU and GPU, AMD GPU, Apple Silicon, and potentially NPU.[39]

scikit-learn : It is a Python module for machine learning, constructed upon SciPy and distributed under the 3-Clause BSD license.[40]

Pandas : is a Python package that provides rapid, flexible, and articulate data structures designed to facilitate straightforward and intuitive manipulation of "relational" or "labelled" data. It aims to function as the fundamental high-level component for conducting practical, real-world data analysis in Python. Its overarching purpose is to establish itself as the most powerful and versatile open-source data analysis and manipulation tool available in any language. Significant advancements are already being made towards this objective.[41]

torch.library : A collection of APIs known as torch.library is utilized to augment PyTorch's fundamental operator library. Besides augmenting operators created with PyTorch's C++ operator registration APIs (such as aten operators), it encompasses tools for testing and designing novel custom operators.[42]

NumPy : NumPy constitutes the essential library for scientific computation in Python. It is a Python library that offers a multidimensional array object, various derived objects (including

[36] https://huggingface.co/docs/transformers/en/index
[37] https://huggingface.co/docs/datasets/en/index
[38] https://huggingface.co/blog/samuellimabraz/peft-methods
[39] https://pypi.org/project/bitsandbytes/
[40] https://pypi.org/project/scikit-learn/
[41] https://pypi.org/project/pandas/
[42] https://docs.pytorch.org/docs/stable/library.html

masked arrays and matrices), and a range of routines for efficient operations on arrays, encompassing mathematical, logical, shape manipulation, sorting, selection, I/O, discrete Fourier transforms, fundamental linear algebra, basic statistical operations, random simulation, and additional functionalities.[43]

Pickle : The pickle module in Python facilitates the serialization of object structures into a byte stream through a process known as pickling, and then enables their deserialization back into objects via unpickling. Although terminology like as serialization, marshalling, or flattening are utilized, "pickling" and "unpickling" are favored for their clarity.[44]

OS : This module enables cross-platform access to operating system features, including file handling with open(), path manipulation via os.path, and command-line file reading using fileinput. For temporary file creation, use tempfile, and for advanced file and directory operations, use shutil.[45]

Hardware Used : The research was performed in a Google Colab environment, which possessed the following attributes: 12.7 GB of RAM accompanied with a T4 GPU with 15 GB of memory.

3.3 Environment and Development tools

Python

Python is a high-level, interpreted programming language characterized by its simplicity and readability, making it ideal for rapid application development and scripting. The inherent data types of Python and its dynamic typing promote modularity and reusability in programs. The edit-test-debug architecture of Python facilitates rapid work generation due to the absence of compilation, and the use of exceptions for error handling simplifies the debugging process. The Python debugger is robust and facilitates the inspection and evaluation of code. The comprehensive standard library is available for free download, and Python is compatible across multiple platforms.[46]

Google colab

Google Colab, developed by Google Research, is an environment akin to Jupyter Notebooks. A Python program developer can utilize this notebook to compose and execute Python programs solely using a web browser. Google Colab is fundamentally a cloud-based Jupyter Notebook. It is unnecessary to install a runtime or upgrade your computer hardware to meet Python's CPU/GPU intensive workload requirements in order to utilize Collab. Colab offers complimentary computational resources, including memory, storage, graphics processing units (GPUs), tensor processing units (TPUs), and processing power. This cloud-based Python coding environment, developed by Google, targets Python students, big data engineers, data scientists, AI researchers, and machine learning developers. The most advantageous aspect is the ability to maintain a single code notebook that encompasses all components necessary for presenting a comprehensive machine learning or data science project to funders or program managers.[47]

[43] https://numpy.org/doc/stable/
[44] https://docs.python.org/3/library/pickle.html
[45] https://docs.python.org/3/library/os.html
[46] https://www.python.org/doc/essays/blurb/
[47] https://geekflare.com/dev/google-colab/

3.4 Datasets

Formspring Dataset

Formspring.me, subsequently rebranded as spring.me, was among the pioneering social question-and-answer platforms targeted towards adolescents. It is a platform that facilitates the exchange of personal information, enables enquiries, and provides responses from other Formspring members or anonymous participants. Comments and enquiries are directed to a private mailbox, where the user may choose to ignore, delete, or react to them. Formspring gained popularity chiefly due to its anonymous features. Adolescents congregated at the site, and the user count exceeded one million in under two months. With the site's increasing popularity, worries around anonymity also escalated. The website incited controversy due to its facilitation of abuse and cyberbullying, leading to its removal in 2013 after several adolescent suicides were associated with bullying on the network. The dataset comprises 50 IDs extracted from Formspring.me in the summer of 2010. The profile information for each ID (BIO, DATE, LOCATION, and USERID) and the message (TEXT: the complete text encompassing the question and answer denoted by the tags "(Q:)" and "(A:)" respectively; ASKER: the ID of the individual posing the question) was extracted. Three annotators from Amazon Mechanical Turk labelled the dataset, which has been periodically updated since its release by Reynolds et al. in 2011. [3]

Impermium Dataset

Impermium provided the second dataset for the Kaggle [21] competition. Impermium is an engineering firm that provides a subscription-based service to assist consumer websites and social media platforms in preventing and eliminating most dubious and abusive transactions. The primary objective of the competition is to ascertain whether a remark made during a public discourse is offensive to another participant in the conversation. The data is structured in a column of labels, where 0 denotes a neutral comment and 1 signifies an insulting comment, accompanied by an attribute field that includes the timestamp of the comment and the content of the comment itself. The collection has 6,580 samples, of which 1,742 are classified as insulting and 4,838 as neutral. In our experiments, we examine the offensive category

Dataset	Class	Size	Total	Examples
Formspring	CyberBullying	1395	9200	Q: ur so fakee .. A: mhmmm..
	No-CyberBullying	7804		Q: Cats or Dogs? A: Dogs! Im allergic 2 cats=
Impermium	CyberBullying	1206	4605	Even as a troll you are a pathetic failure
	No-CyberBullying	3399		Oh cool some more internet points for your collection

Table 3.1 Distribution of classes and examples of the datasets used

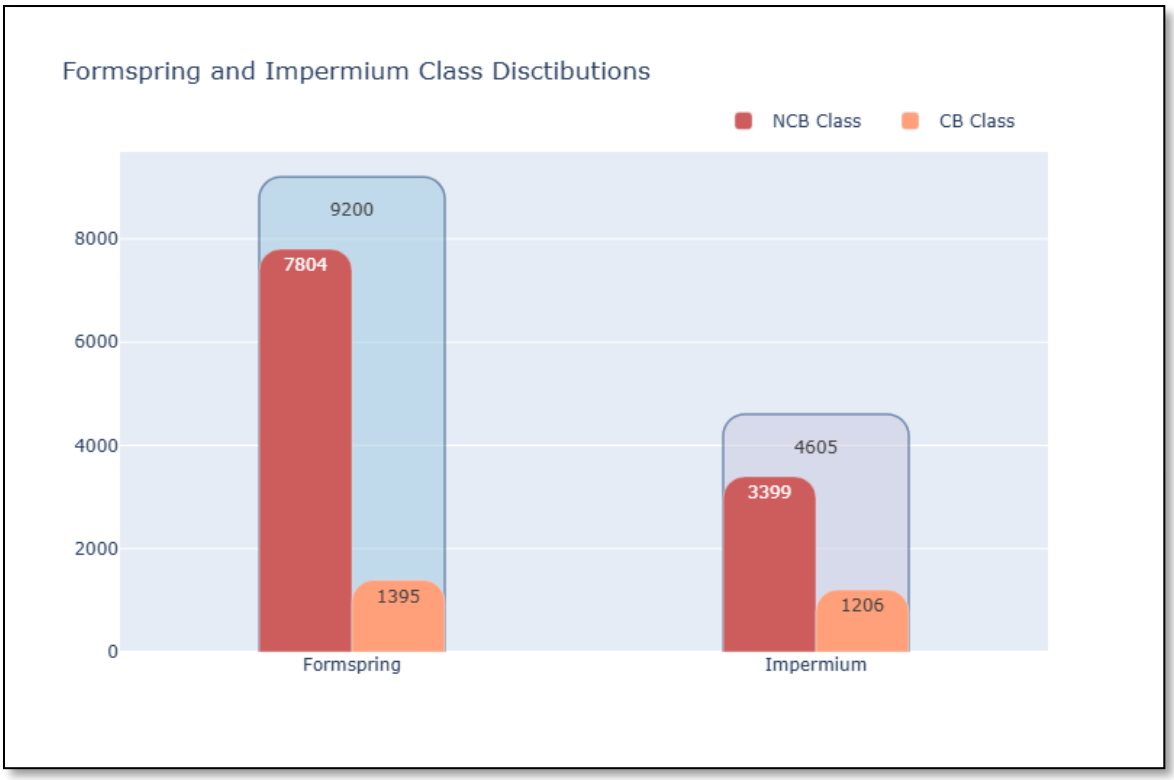


Figure 3.2 Formspring and Impermium Class Distributions

3.4.1 Pre-Processing

Social media and user-generated content are often chaotic, unreliable, and fragmented. To render this data comprehensible and suitable for classification, various preprocessing techniques are necessary.

Formspring Dataset: This dataset comprises posts that feature text marked with "Q:" for queries and "A:" for answers. The initial phase involves eliminating these identifiers, succeeded by a lowercase transformation that eradicates various forms of noise from the text, encompassing HTML components, special characters, and punctuation.

Impermium Dataset: RAKIB and SOON (2018)[48] posted a preprocessed version of this dataset to their GitHub repository. The text cleaning procedure involved the removal of hyperlinks, the normalization of words by regular expressions (e.g., Looolll -> Looll), the standardization of phrases (e.g., you are -> you are), and the elimination of non-alphabetic characters and lowercase letters.

3.4.2 Splitting the data

Dataset	Train	Test	Valid
Formspring	5520	1840	1840
Impermium	2763	921	921

Table 3.2 Splitting the data

The datasets were partitioned into 60% for training and 20% for testing , while the remaining 20% is reserved for validation.

[48] https ://github.com/tazeek/BullyDetect

3.5 Proposed method for unbalanced Datasets :

Class imbalance is a common issue in cyberbullying detection tasks, characterized by a predominance of samples from the non-cyberbullying class, whereas instances of cyberbullying are under-represented. To address this problem in the Impermium and Formspring datasets, two loss functions are utilized: weighted Cross-Entropy Loss and Focal Loss.

Weighted Cross-Entropy Loss alters the conventional cross-entropy formulation by incorporating class-specific weights. These weights augment the penalty for misclassifying minority class instances, hence prompting the model to focus more on under-represented cases during training. Weights are often established according to the inverse frequency of each class within the dataset, so ensuring that errors in seldom classes exert a greater impact on the loss and subsequent parameter adjustments.

Focal Loss is intended to mitigate class imbalance by dynamically adjusting the loss for each sample according to the model's confidence level. It diminishes the influence of well-classified (easy) examples and accentuates challenging ones, which are more likely to pertain to the minority class. This is accomplished by employing a modulating factor that reduces the loss for confident predictions, so redirecting attention to instances that the model finds challenging to categorize accurately.

The implementation of these two strategies seeks to enhance classification performance, especially for the minority class (cyberbullying), by rectifying the imbalance at the loss function level. This method improves recall and F1-score for the under-represented class, resulting in more balanced and resilient model predictions.

3.6 Models Fine tuning with PEFT (LoRA)

Large Language Models (LLMs), including GPT, LLaMA, and Mistral, have exhibited exceptional proficiency in many natural language processing (NLP) tasks. Nonetheless, the traditional method of complete fine-tuning—wherein all model parameters are modified—presents considerable computational and resource difficulties. Due to the multitude of parameters in contemporary LLMs, comprehensive fine-tuning necessitates significant GPU resources, ample memory, extended training durations, and big annotated datasets. These constraints make the approach excessively costly and unattainable for numerous researchers and practitioners.

A set of approaches termed Parameter-Efficient Fine-Tuning (PEFT) has emerged to address these limitations. PEFT approaches seek to tailor pre-trained LLMs for downstream tasks by altering just a limited number of parameters, thus minimizing computational and memory demands. [4]

Among the diverse PEFT techniques, Low-Rank Adaptation (LoRA) has attained notable attention owing to its efficiency and effectiveness. LoRA functions by maintaining the pre-trained model parameters in a static state while integrating supplementary low-rank matrices into the attention or feedforward layers. Only the newly introduced parameters are tuned during training. This markedly decreases the quantity of trainable parameters while maintaining the original model's expressive capacity. [5]

This study selects LoRA for numerous crucial reasons:

- Computational Efficiency: LoRA significantly reduces memory consumption and training duration in comparison to complete fine-tuning.

- Architectural Compatibility: It seamlessly integrates with current transformer-based models and widely-used frameworks such as Hugging Face Transformers.
- Empirical Performance: Despite a diminished parameter footprint, LoRA regularly attains competitive, and frequently state-of-the-art, performance across many NLP benchmarks.
- The approach facilitates the fine-tuning of extensive models (e.g., Mistral-7B) with limited hardware resources.

Quantization techniques are utilized in conjunction with LoRA to improve computing efficiency. Quantization diminishes the precision of the model's weights, typically transitioning from 16-bit or 32-bit floating point to lower-bit forms like 8-bit or 4-bit integers, which substantially reduces memory usage and enhances inference and training speed. Specifically, 4-bit quantization has demonstrated significant efficacy in minimizing hardware demands while preserving satisfactory performance standards. When combined with LoRA, quantization facilitates the fine-tuning and deployment of LLMs in resource-constrained contexts, such as consumer-grade GPUs, without significant loss in model fidelity. This amalgamation renders cutting-edge language models available for a wider array of applications and users.

3.7 Models used

This study ensures a comprehensive evaluation of performance, efficiency, and scalability across several natural language processing applications by employing a range of cutting-edge language models:

Mistral 7B : The advancement of lightweight, high-performing large language models (LLMs) is progressing swiftly as researchers endeavor to reconcile computing efficiency with linguistic proficiency. Mistral 7B is a significant advancement in this domain, with a compact architecture with seven billion parameters that provides exceptional performance and facilitates implementation on a broader array of hardware platforms. It integrates two principal innovations—sliding window attention (SWA) and grouped-query attention (GQA). SWA enhances the management of extended sequences by focusing on a fixed-size context window, hence decreasing inference expenses, whereas GQA refines multi-head attention to provide more efficient processing. The design choices enable Mistral 7B to execute natural language understanding and generating jobs with enhanced speed and scalability. [6]



Figure 3.3 Mistral Model.[49]

[49] https://www.free-work.com/fr/tech-it/blog/actualites-informatiques/mistral-ai-la-nouvelle-deferlante-ia

Empirical research indicates that Mistral 7B trumps larger models like LLaMA 2 13B and even LLaMA 1 34B in certain domains, illustrating that architectural efficiency can match or exceed the advantages of a higher parameter count. The release of Mistral 7B - Instruct, a fine-tuned version for instruction-following tasks, surpasses LLaMA 2 13B - Chat in both human and automated evaluations, demonstrating its efficacy for interactive, task-oriented applications. Importantly, both the base and instruction-tuned variants are open-sourced under the permissive Apache 2.0 license, facilitating wider deployment in academic and commercial contexts. [6]

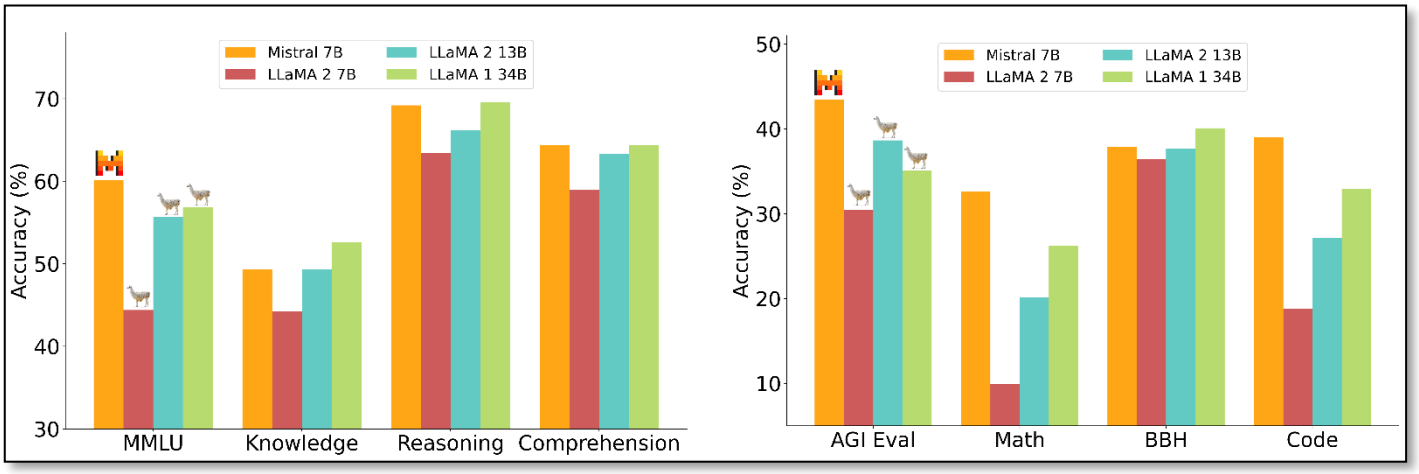


Figure 3.4 Comparison Mistral 7B with Llama2 Family.[50]

Model	Modality	MMLU	HellaSwag	WinoGrande	PIQA	Arc-e	Arc-c	NQ	TriviaQA	HumanEval	MBPP	MATH	GSM8K
LLaMA 2 7B	Pretrained	44.4%	77.1%	69.5%	77.9%	68.7%	43.2%	24.7%	63.8%	11.6%	26.1%	3.9%	16.0%
LLaMA 2 13B	Pretrained	55.6%	**80.7%**	72.9%	80.8%	75.2%	48.8%	**29.0%**	**69.6%**	18.9%	35.4%	6.0%	34.3%
Code LLaMA 7B	Finetuned	36.9%	62.9%	62.3%	72.8%	59.4%	34.5%	11.0%	34.9%	**31.1%**	**52.5%**	5.2%	20.8%
Mistral 7B	Pretrained	**60.1%**	**81.3%**	**75.3%**	**83.0%**	**80.0%**	**55.5%**	**28.8%**	**69.9%**	**30.5%**	47.5%	**13.1%**	**52.1%**

Table 3.3 Benchmarks by their themes.[50]

Mistral 7B is an exemplary option for practical and accessible implementation in various real-world language applications because to its robust performance, computational efficiency, scalability, and open license.

Llama2 7B : Recent developments in large language models (LLMs) have resulted in the introduction of LLaMA 2, a collection of open-source models varying from 7 to 70 billion parameters, designed for both foundational and conversational applications. The LLaMA 2 7B model, as described by Touvron et al. (2023), is optimized for discussion tasks using reinforcement learning from human feedback (RLHF) and safety-focused methodologies, establishing it as a vital asset for research in open and responsible AI. [7]

Subsequent research has broadened its functionalities across other sectors. Chen et al. (2023) present LongLoRA, a low-rank adaptation technique that facilitates efficient fine-tuning for long-context inputs while incurring minimal computing costs. Pang et al. (2024) present a phased instruction tuning architecture that enhances the model's generalization and instruction-following capabilities via progressively intricate training challenges. [8]

[50] https://mistral.ai/news/announcing-mistral-7b

In the biomedical field, Tran et al. (2023) implement domain-specific instruction tailoring with a tailored biological dataset, showcasing the adaptability and enhanced efficacy of LLaMA 2 7B on specialized NLP tasks. [9]



Figure 3.5 Llama Model Family.[50]

These works collectively demonstrate the adaptability and resilience of LLaMA 2 7B across several applications, including general-purpose dialogue and domain-specific language processing in fields such as healthcare and education. LLaMA 2 7B, with its robust basis, shown adaptability via fine-tuning methodologies, and dedication to ethical AI methods, is an attractive option for both general and niche natural language processing applications. [10]

Llama3.2 1B : The advancement of LLaMA 3.2 signifies a significant progression in large language model (LLM) architecture, delivering robust performance across diverse jobs while ensuring efficiency. [11]



Figure 3.6 Llama 3.2 Family.[51]

[50] https://www.llama.com/llama2/
[51] https://ai.meta.com/blog/llama-3-2-connect-2024-vision-edge-mobile-devices/

Grattafiori et al. (2024) introduced these foundational models as part of the LLaMA 3 series, which includes a 1B parameter variation, optimized for multilingual activities, reasoning, programming, and tool utilization. LLaMA 3.2 is engineered for versatility, proficient in managing tasks related to natural language comprehension, logical reasoning, and computing. Empirical assessments reveal its competitive efficacy across multiple benchmarks, including comparisons with premier models such as GPT-4. [11]

Pretrained Models			
Capability	**Benchmark**	**Llama 3.2 1B**	**Llama 3.2 3B**
General	MMLU 5 shot	32.2	58
General	AGIEval English 3-5 shot	23.3	39.2
General	ARC-Challenge 25 shot	32.8	69.1
Reading Comprehension	SQuAD 1 shot	49.2	67.7
Reading Comprehension	QuAC 1 shot (F1)	37.9	42.9
Reading Comprehension	DROP 3 shot (F1)	28.0	45.2
Long Context	Needle in Haystack	96.8	1
Instruction Tuned Models			
Capability	**Benchmark**	**Llama 1B**	**Llama 3B**
General	MMLU 5 shot	49.3	63.4
Re-writing	Open-rewrite eval	41.6	40.1
Summarization	TLDR9+ 1 shot (test)	16.8	19.0
Instruction following	IFEval	59.5	77.4
Math	GSM8K 8 shot (CoT)	44.4	77.7
Math	MATH (CoT)	30.6	48.0
Reasoning	ARC-C	59.4	78.6
Reasoning	GPQA	27.2	32.8
Reasoning	Hellaswag	41.2	69.8
Tool Use	BFCL V2 (AST Summary)	25.7	67.0
Tool Use	Nexus	13.5	34.3
Long Context	InfiniteBench/En.QA	20.3	19.8
Long Context	InfiniteBench/En.MC	38.0	63.3
Multilingual	NIH/Multi-needle	75.0	84.7
Multilingual	MGSM (CoT)	24.5	58.2

Table 3.4 Llama 3.2 1B & 3B Benchmarks.[52]

LLaMA 3.2 transcends textual applications by including multimodal functionalities, adeptly executing tasks in picture recognition, video comprehension, and audio analysis via a compositional, cross-modal methodology. Although post-training versions remain under development, the pre-trained models have been released to the public, improving accessibility for study and application.

SmolLM2 1.7B : The rising demand for efficient and proficient language models has resulted in the creation of smaller-scale systems that compete with bigger models in performance. SmolLM2 1.7B, presented by Ben Allal et al. (2024), shows this advancement through a data-centric methodology that facilitates performance comparable to or surpassing models such as Qwen2.5-1.5B and LLaMA3.2-1B. SmolLM2, developed using a varied 11 trillion-token corpus via a multi-stage pipeline, integrates specialized datasets—FineMath, Stack-Edu, and SmolTalk—to improve mathematical reasoning, instructional comprehension, and conversational ability. Strategic enhancements during training and optimized dataset

[52] https://www.reddit.com/r/LocalLLaMA/comments/1fp9wem/llama_32_1b_3b_benchmarks/

integration substantially enhanced generalization and output quality, as confirmed by ablation investigations. [12]

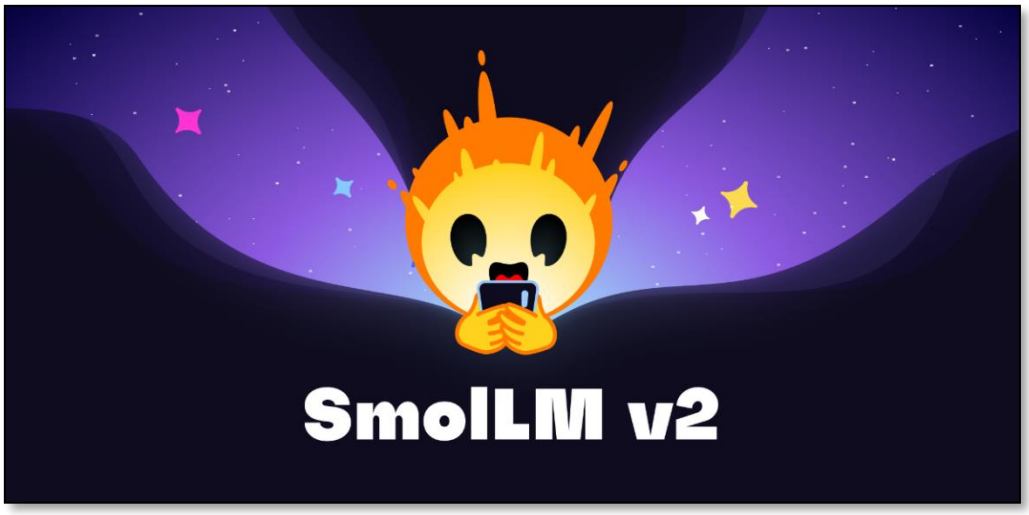


Figure 3.7 SmolLM2 Model.[53]

Metric	**SmolLM2-1.7B**	**Llama-1B**	**Qwen2.5-1.5B**	**SmolLM1-1.7B**
HellaSwag	**68.7**	61.2	66.4	62.9
ARC (Average)	**60.5**	49.2	58.5	59.9
PIQA	**77.6**	74.8	76.1	76.0
MMLU-Pro (MCF)	**19.4**	11.7	13.7	10.8
CommonsenseQA	**43.6**	41.2	34.1	38.0
TriviaQA	**36.7**	28.1	20.9	22.5
Winogrande	**59.4**	57.8	59.3	54.7
OpenBookQA	42.2	38.4	40.0	**42.4**
GSM8K (5-shot)	31.0	7.2	**61.3**	5.5

Table 3.5 evaluation results of SmolLM2 vs other models . [53]

In conjunction with this research, Alrashed (2024) examines instruction tuning in SmolTulu-1.7B-Instruct, a refined version of SmolLM2 developed utilizing AllenAI's Tulu 3 framework. The research indicates that adjusting learning rate and batch size ratios substantially influences task performance—lower ratios benefit pattern recognition tasks, whereas larger ratios improve reasoning abilities. [13]

[53] https://huggingface.co/HuggingFaceTB/SmolLM2-1.7B

Collectively, these contributions illustrate that with meticulous data design and flexible training techniques, compact models like as SmolLM2 may attain competitive outcomes across a wide array of NLP tasks while maintaining resource efficiency and accessibility.

Falcon3 1B : The swift advancement of large language models (LLMs) has catalyzed an increase in open-source projects focused on providing superior performance with extensive accessibility. The Falcon model series by Almazrouei et al. (2023) is particularly notable. The autoregressive transformer models, including Falcon 1B, 7B, and the flagship Falcon 180B, were trained on over 3.5 trillion high-quality tokens predominantly obtained from the internet. The research highlights the scalability of Falcon's architecture and its novel training process, facilitating both extensive and lightweight deployment without sacrificing model quality.



Figure 3.8 Falcon3 Model.[54]

Falcon 180B competes with proprietary models like as PaLM-2 Large in language comprehension and generation, while incurring lower training and inference expenses. The architecture and data pipeline utilized for the larger models are also employed for smaller variations such Falcon 3B and 1B, rendering them appropriate for edge applications while ensuring consistency in performance and quality. The models were trained on a cloud-optimized infrastructure utilizing up to 4,096 NVIDIA A100 GPUs, with bespoke distributed techniques to address bandwidth limitations, hence assuring efficient and reproducible large-scale training. In alignment with open science principles, the developers have unveiled the Falcon 7B, 40B, and 180B models, accompanied by a 600 billion token subset of their dataset. This dedication promotes transparency, reproducibility, and innovation within the open-source LLM community. [14]

The Falcon model family is a strategic option for both high-resource implementations and lightweight, accessible AI solutions due to its superior performance across various model sizes, fast and scalable training approach, and strong dedication to open research.

[54] https://huggingface.co/tiiuae/Falcon3-1B-Base

3.8 Conclusion

Chapter 3 elucidates the Large Language Model (LLM) methodology and technology employed in the development of a cyberbullying detection system. The chapter describes Formspring and Impermium, which exhibit an imbalance between cyberbullying and non-cyberbullying content. these datasets underwent meticulous preprocessing to eliminate HTML elements and special characters. This paper tackled class imbalance using a class-weighted cross-entropy and Focal loss. The methodology emphasizes underrepresented instances of cyberbullying by inversely weighting classes based on their frequency, hence enhancing detection accuracy and reducing bias towards majority groups.

The chapter presents advanced LLMs Mistral 7B, Llama2 7B, Llama3.2 1B, SmolLM2 1.7B, and Falcon3 1B, selected for their architectural merits. The Google Colab environment utilizing T4 GPU employed Hugging Face Transformers and PEFT to facilitate model adaptation and assessment.

The binary classification pipeline employs contextual embeddings from large language models and to rigorously examine linguistic and semantic patterns. Dividing the datasets into training (60%), validation (20%), and test (20%) sets facilitated comprehensive performance evaluation. In the following chapter, the results of fine-tuning previously mentioned LLMs on cyberbullying classification will be discussed.

Chapter 4: Results & Discussion

4.1 Introduction

Chapter 4 presents the empirical results of Chapter 3's Large Language Model (LLM)-based cyberbullying detection system. From methodological design to critical assessment, this chapter examines model efficacy, comparative evaluation, and practical implications. This chapter thoroughly evaluates the five LLMs discussed in chapter 3 on cyberbullying detection using the Formspring and Impermium datasets. Precision, recall, F1-score, and computational efficiency evaluate the models' capacity to handle imbalanced data and contextual complexities in user-generated text.

This chapter evaluates LLM-driven methodologies in comparison to leading baselines, situating the analysis within the wider context of cybersecurity and ethical artificial intelligence. Critical issues, including false positives, cultural bias inherent in training data, and the scalability of models for practical application, are thoroughly analyzed. The results demonstrate the practical use of theoretical advancements, including Focal Loss and parameter-efficient tuning procedures, to improve cyberbullying detection systems.

4.2 Models Training Parameters

The parameters utilized in the cyberbullying detection method are examined, accompanied by a rationale for the selected values to guarantee optimal performance.

Model & Quantization Parameters :

Quantization is a method employed in machine learning to diminish the size and computational expense of models by transforming high-precision numbers (such as 32-bit floats) into lower-precision formats (such 8-bit integers or 4-bit values). This minimizes memory consumption and accelerates inference, enhancing model efficiency for deployment on resource-constrained devices. Although quantization may somewhat diminish accuracy, meticulous execution can preserve performance while enhancing efficiency.

Quantization Facilitates the fine-tuning of a substantial models on constrained memory (e.g., Colab 15GB) while maintaining satisfactory accuracy.

Parameter	Value	Rationale
num_labels	2	Binary classification task → CB vs. NCB.
torch_dtype	torch.float16	Conserves memory and enhances speed on GPUs that accommodate half-precision (e.g., Colab's T4/V100).
load_in_4bit	True	Quantizes model weights to 4 bits, significantly decreasing memory use.
bnb_4bit_compute_dtype	torch.float16	Computation remains in FP16 to achieve an optimal equilibrium between speed and precision.

bnb_4bit_use_double_quant	True	Introduces an additional quantization level to enhance precision at 4-bit.
bnb_4bit_quant_type	nf4	NormalFloat4 is a superior quantization approach for large language models compared to fp4.

Table 4.1 Model & Quantization Parameters

```
# qunatization config
quantization_config = BitsAndBytesConfig(
    load_in_4bit = True, # enable 4-bit quantization
    bnb_4bit_quant_type = 'nf4', # information theoretically optimal dtype for normally distributed weights
    bnb_4bit_use_double_quant = True, # quantize quantized weights //insert xzibit meme
    bnb_4bit_compute_dtype = torch.bfloat16 # optimized fp format for ML
)
```

Figure 4.1 Model & Quantization Parameters

LoRA Configuration :
Enhances memory efficiency and training velocity by adjusting only minor adapters rather than the entire model. Optimal for constrained GPU RAM and diminutive datasets.

Parameter	Value	Rationale
r	16	The rank of the LoRA matrices governs parameter efficiency.
lora_alpha	32	Scaling coefficient for the LoRA modification. Higher values may enhance learning, albeit at the expense of stability.
target_modules	["q_proj","k_proj","v_proj","o_proj"]	Only refine the query and value projection layers, which are very efficacious in transformer models.
lora_dropout	0.05	Mitigates overfitting by randomly omitting 5% of LoRA weights during the training process.
bias	"none"	Maintains bias terms in a fixed state to diminish training complexity.

Table 4.2 LoRA Configuration

```
# lora config
lora_config = LoraConfig(
    r = 16, # the dimension of the low-rank matrices
    lora_alpha = 8, # scaling factor for LoRA activations vs pre-trained weight activations
    target_modules = ['q_proj', 'k_proj', 'v_proj', 'o_proj'],
    lora_dropout = 0.05, # dropout probability of the LoRA layers
    bias = 'none', # wether to train bias weights, set to 'none' for attention layers
    task_type = 'SEQ_CLS'
)
```

Figure 4.2 LoRA Configuration

TrainingArguments :

The Hugging Face transformers library offers a configuration class called TrainingArguments, which contains the collection of runtime parameters and hyperparameters needed for model evaluation, checkpointing, and training. It streamlines experiment management and allows for the control of several training process elements.

Parameter	Value	Rationale
num_train_epochs	3	Sufficient for convergence on smaller datasets such as Impermium and Formspring.
per_device_train_batch_size gradient_accumulation_steps	2 4	Simulates an optimal batch size of 2 × 4 = 8, ensuring equilibrium between memory utilization and stability.
warmup_steps	10	Gradually escalates the learning rate initially, so averting abrupt divergence.
weight_decay	0.01	Implementation of regularization techniques to mitigate overfitting.
learning_rate	2e-4	Marginally elevated compared to the default for LoRA, as LoRA layers require an increased learning rate to facilitate rapid learning while the remainder of the model remains static.
save_total_limit	1	Retains solely the optimal model → preserves disk space.

load_best_model_at_end	True	Automatically reinstates the model that achieved the highest F1 score after validation.
metric_for_best_model	"f1"	F1 is a superior statistic compared to accuracy for imbalanced binary classification.

Table 4.3 TrainingArguments

```
training_args = TrainingArguments(
    output_dir="llama2",
    learning_rate=lr,
    lr_scheduler_type= "constant",
    warmup_ratio= 0.1,
    max_grad_norm= 0.3,
    per_device_train_batch_size=batch_size,
    per_device_eval_batch_size=batch_size,
    num_train_epochs=num_epochs,
    weight_decay=0.01,
    evaluation_strategy="epoch",
    save_strategy="epoch",
    load_best_model_at_end=True,
    report_to="none",
    fp16=True,
    gradient_checkpointing=True,
)
```

Figure 4.3 TrainingArguments

Loss Function (Weighted Cross-Entropy Loss and Focal Loss) :
Mitigates model bias towards the majority class (e.g., "not Cyberbullying") by prioritizing the minority class ("Cyberbullying").

Parameter	Value	Rationale
loss_fct	nn.CrossEntropyLoss(weight=class_weights)	Calculated utilizing compute_class_weight("balanced", ...) → addresses class imbalance.
alpha	[0.3, 0.7]	Class imbalance: The dataset has more negative (class 0) examples than positive (class 1), so alpha emphasizes the minority (positive) class. A higher weight (0.7) is given to class 1 to reduce bias toward the majority class.

gamma	2.0	Focusing parameter: Standard value used to reduce the impact of easy (well-classified) examples and focus the loss on hard (misclassified) examples. Helps the model learn from challenging cases.

Table 4.4 Loss Function

```
class WeightedCELossTrainer(Trainer):
    def compute_loss(self, model, inputs, return_outputs=False):
        labels = inputs.pop("labels")
        # Get model's predictions
        outputs = model(**inputs)
        logits = outputs.get("logits")
        # Compute custom loss
        loss_fct = torch.nn.CrossEntropyLoss(weight=torch.tensor([neg_weights, pos_weights], device=model.device, dtype=logits.dtype))
        loss = loss_fct(logits.view(-1, self.model.config.num_labels), labels.view(-1))
        return (loss, outputs) if return_outputs else loss
```

Figure 4.4 Weighted Cross-Entropy Loss Parameters

```
class FocalLoss(torch.nn.Module):
    def __init__(self, alpha=0.25, gamma=2.0, reduction='mean'):
        super(FocalLoss, self).__init__()
        self.alpha = alpha
        self.gamma = gamma
        self.reduction = reduction
        self.ce = torch.nn.CrossEntropyLoss(reduction='none')

    def forward(self, inputs, targets):
        logpt = -self.ce(inputs, targets)
        pt = torch.exp(logpt)
        focal_term = (1 - pt) ** self.gamma

        # Handle alpha if it's a tensor (e.g., class weights)
        if isinstance(self.alpha, torch.Tensor):
            at = self.alpha.gather(0, targets)
        else:
            at = self.alpha

        loss = -at * focal_term * logpt

        if self.reduction == 'mean':
            return loss.mean()
        elif self.reduction == 'sum':
            return loss.sum()
        return loss
```

Figure 4.5 Focal loss parameters

4.3 Results of Impermium / Formsprimg datasets:

The results delineates and contrasts the efficacy of five fine-tuned language models discussed in chapter 3 utilizing the Impermium and formspring datasets for the classification of Cyberbullying comments in two methods.

Classification Reports :

The classification report offers a comprehensive assessment of each model's efficacy on the Impermium and formspring datasets, emphasizing measures including precision, recall, and F1-score for both categories (Cyberbullying and non-Cyberbullying). These measures are essential for evaluating the efficacy of each model in addressing the class imbalance present in the dataset. Performance variations were noted among the five analyzed models. Models such as Mistral 7B and LLaMA 2 7B HF typically shown superior recall and F1-scores, signifying enhanced proficiency in detecting harmful comments while minimizing false negatives. Conversely, smaller models like Falcon 3 1B and LLaMA 3.2 1B had relatively inferior performance, especially in identifying harmful situations, frequently prioritizing recall above precision. The classification report functions as an essential instrument for evaluating the trade-offs among various models, facilitating the identification of the most efficient architecture for practical Cyberbullying comment detection applications.

		Formspring			**impermium**		
		Metrics					
	Models	**Precision**	**Recall**	**F1-score**	**Precision**	**Recall**	**F1-score**
Cyberbullying	Mistral 7B	0.29	0.87	0.44	0.78	0.76	0.77
	LLaMA2 7B	0.44	0.75	0.55	0.70	0.89	0.78
	LLaMA3.2	0.59	0.55	0.57	0.71	0.87	0.78
	Falcon 3	0.45	0.71	0.55	0.80	0.74	0.77
	SmolLM2	0.43	0.75	0.54	0.79	0.79	0.79
Non Cyberbullying	Mistral 7B	0.96	0.59	0.73	0.91	0.92	0.92
	LLaMA2 7B	0.94	0.81	0.87	0.96	0.86	0.90
	LLaMA3.2	0.91	0.92	0.92	0.95	0.87	0.91
	Falcon 3	0.94	0.83	0.88	0.91	0.93	0.92
	SmolLM2	0.94	0.80	0.87	0.92	0.92	0.92

Table 4.5 Model results on the two datasets Formspring and Impermium in ***Weighted Cross-Entropy Loss Method***

		Formspring			impermium		
		Metrics					
	Models	**Precision**	**Recall**	**F1-score**	**Precision**	**Recall**	**F1-score**
Cyberbullying	Mistral 7B	0.89	0.34	0.49	0.78	0.76	0.77
	LLaMA2 7B	0.39	0.76	0.51	0.87	0.62	0.72
	LLaMA3.2	0.74	0.38	0.50	0.86	0.66	0.75
	Falcon 3	0.52	0.62	0.57	0.78	0.77	0.78
	SmolLM2	0.48	0.70	0.57	0.76	0.80	0.78
Non Cyberbullying	Mistral 7B	0.89	0.99	0.94	0.89	0.97	0.93
	LLaMA2 7B	0.94	0.76	0.84	0.87	0.97	0.92
	LLaMA3.2	0.89	0.97	0.93	0.88	0.96	0.92
	Falcon 3	0.92	0.89	0.91	0.92	0.92	0.92
	SmolLM2	0.94	0.85	0.89	0.92	0.90	0.91

Table 4.6 Model results on the two datasets Formspring and Impermium in ***Focal Loss Method***

Confusion Matrices:

The confusion matrix offers a visual and quantitative depiction of the classification results for each model on the Impermium and formspring dataset. It presents the quantities of true positives, true negatives, false positives, and false negatives, facilitating a clear comprehension of how each model differentiates between cyberbullying and non-cyberbullying comments. The confusion matrix for each model elucidates patterns of misclassification, including a propensity to categorize cyberbullying comments as non-cyberbullying or the reverse. This insight is crucial for datasets exhibiting class imbalance, as it indicates whether a model exhibits bias towards forecasting the dominant class. Analyzing these matrices allows for an assessment of each model's predictive efficacy beyond mere accuracy, providing profound insight into model dynamics and the influence of fine-tuning methodologies on practical cyberbullying detection endeavors.

Weighted Cross-Entropy Loss Method Results :

- **Impermium Dataset :**

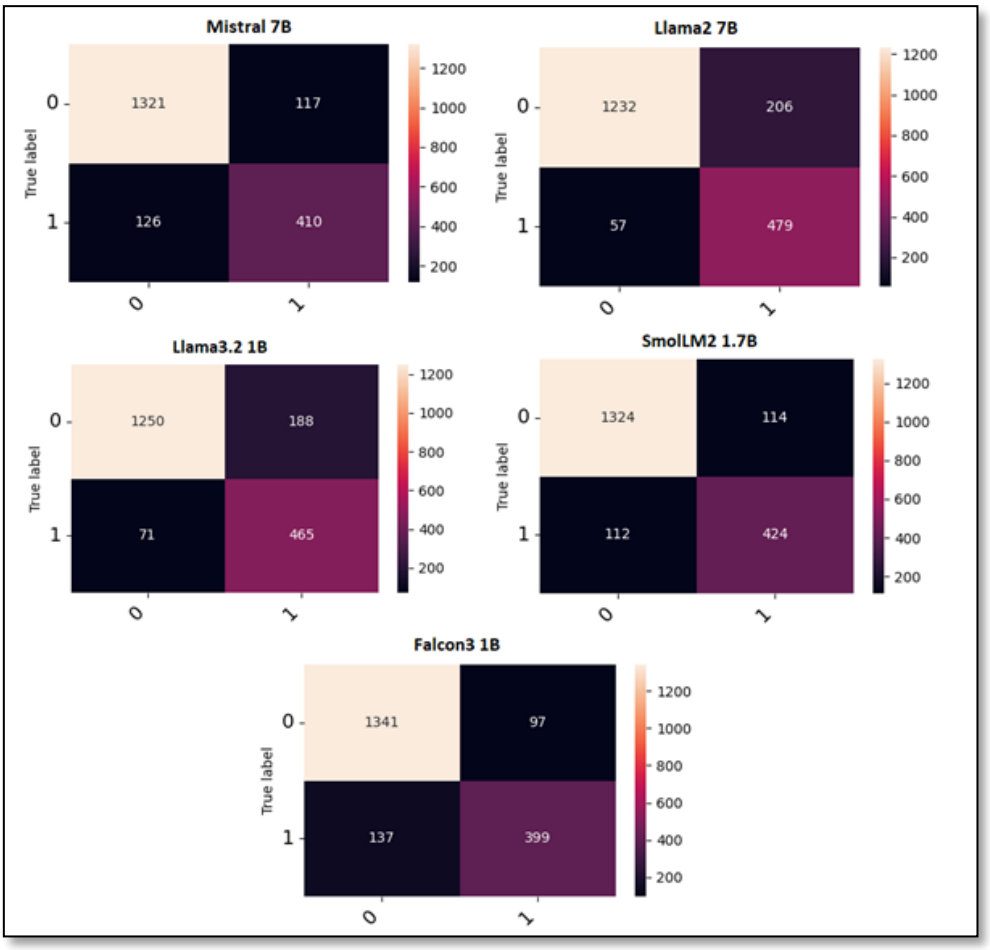


Figure 4.6 Confusion Matrices Impermium Dataset in ***Weighted Cross-Entropy Loss Method***

The confusion matrices for the five models show that binary classification works differently for each one. Llama2 7B is great at finding true positives, with 479 correct predictions, but it has a higher false positive rate of 206. On the other hand, Falcon 3B is great at finding false positives, with only 97, and it has the highest true negatives, with 1341. However, it has the lowest true positives, with only 399, and the highest false negatives, with 137, which means it has a lower recall. Llama3.2 1B does better with true negatives, getting 1250, but it has fewer true positives (465) and more false positives (188) than Llama2 7B, so it doesn't have a clear advantage. Both SmolLM 2.7B and Mistral 7B do well, with SmolLM getting 1324 true negatives and 424 true positives right and Mistral getting 1321 true negatives and 410 true positives right, both keeping their false positives low at 114 and 117, respectively. Overall, Falcon 3B is optimal for decreasing false positives, Llama2 7B for maximizing true positives, and SmolLM 2.7B or Mistral 7B for a balanced approach, depending on whether precision, recall, or overall balance is emphasized.

- **Formspring Dataset :**

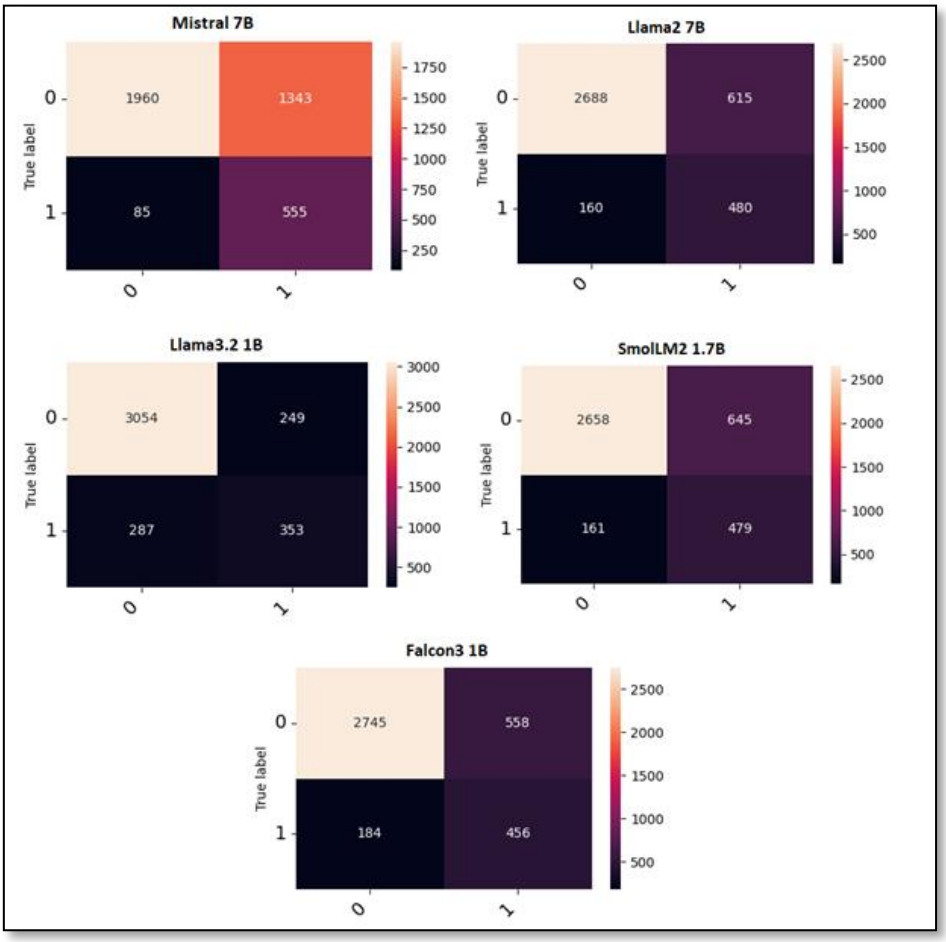


Figure 4.7 Confusion Matrices Formspring Dataset in ***Weighted Cross-Entropy Loss Method***

The confusion matrices for the five models on the Formspring dataset indicate that each model performs binary classification (labels 0 and 1) in a distinct manner. Falcon3 1B exhibits a balanced performance, recording 2745 true negatives (TN) and 456 true positives (TP), although it also demonstrates a moderate incidence of misclassifications, with 578 false positives (FP) and 184 false negatives (FN). Llama2 7B demonstrates superior performance in identifying true negatives (TN) at 2688 and true positives (TP) at 480 compared to Falcon3; nevertheless, it exhibits a higher false positive (FP) rate of 615, indicating a propensity for overpredicting positive outcomes. Llama3.2 1B excels in predicting TN (3054), but does poorly in predicting TP (353). It also exhibits several false positives (249) and false negatives (287), indicating difficulty in accurately identifying positive classifications. Mistral 7B possesses the highest FP (1343), resulting in significantly diminished accuracy. It possesses a commendable true negative (TN) rate of 1960 and a true positive (TP) rate of 555, alongside the lowest false negative (FN) rate of 85, indicating a minimal likelihood of overlooking positive instances, however with the potential for overprediction. SmolLM2 1.7B closely parallels Llama2 7B, exhibiting 2658 true negatives, 479 true positives, 645 false positives, and 161 false negatives, indicating a balanced albeit unremarkable performance. Llama3.2 1B excels in minimizing false positives, but Mistral 7B exhibits a propensity for over-predicting

positives, often inaccurately. Llama2 7B and SmolLM2 1.7B occupy an intermediate position regarding balance.

Focal Loss Method Results :

- **Impermium Dataset :**

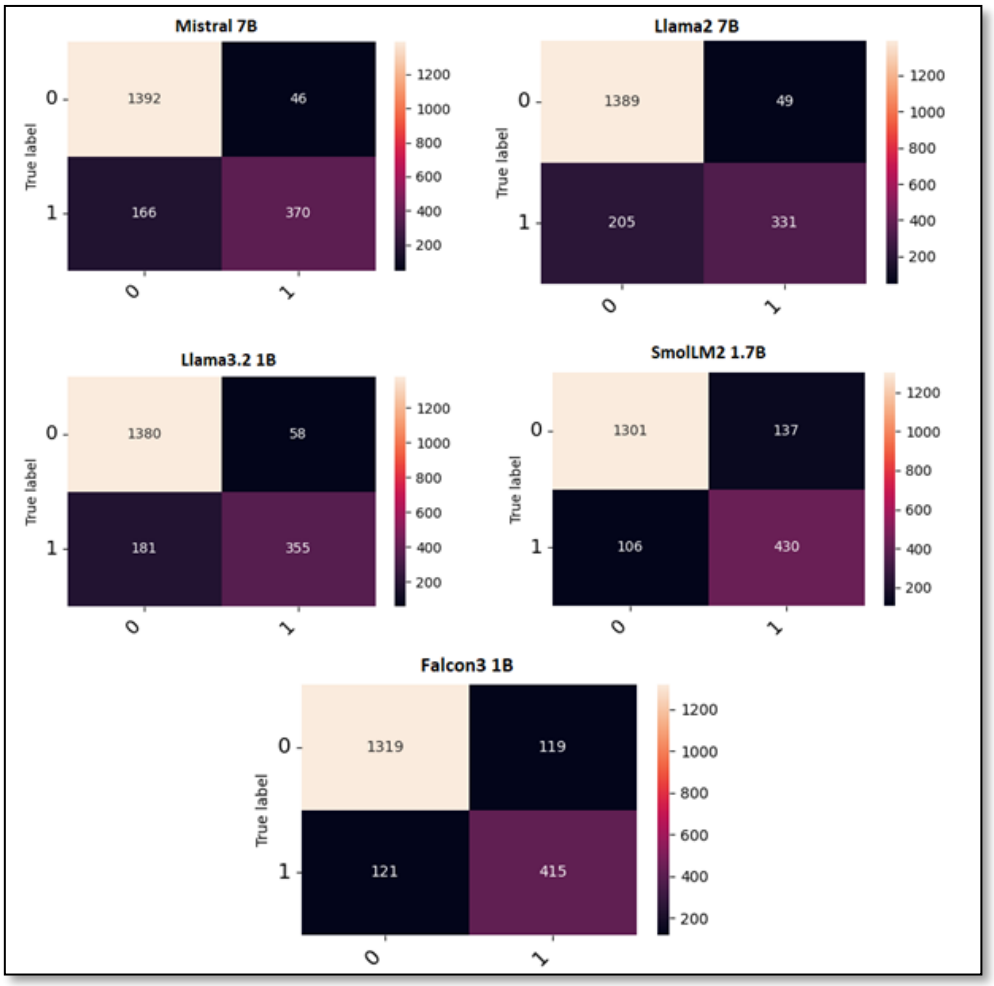


Figure 4.8 Confusion Matrices Impermium Dataset in ***Focal Loss Method***

The confusion matrices show how well the Mistral 7B, Llama2 7B, Llama3.2 1B, SmolLM2 1B, and Falcon3 1B models did on the Impermium Dataset using the Focal Loss approach. They compare predicted labels (0 or 1) with genuine labels (0 or 1) and provide the number of each outcome. Mistral 7B has 1392 true negatives, 46 false positives, 164 false negatives, and 373 real positives, which means it works well overall but misses a lot of positive predictions. Llama2 7B gets 1399 true negatives, 49 false positives, 205 false negatives, and 331 real positives. This shows a comparable conservative bias but a higher number of erroneous negatives. Llama3.2 1B has 1330 true negatives, 56 false positives, 181 false negatives, and 355 true positives. It has a modest false negative rate and a good true positive count, but it has less true negatives. SmolLM2 1B has 1301 true negatives, 137 false positives, 106 false negatives, and 430 true positives. It does better on true positives and has fewer false negatives, but it has a larger false positive rate. Falcon3 1B has 1319 true negatives, 119 false positives, 121 false negatives, and 415 true positives. This means that it performs well overall, with fewer false negatives and a strong true positive count, even though it has a lot of false positives. In

general, bigger models like Mistral 7B and Llama2 7B tend to focus on true negatives but miss positive cases. Smaller models like SmolLM2 1B and Falcon3 1B make it easier to find positives, but they also make it easier to find false positives. The Focal Loss method seems to shift focus to harder examples, which means that there are trade-offs between precision and recall across the models.

Formspring Dataset :

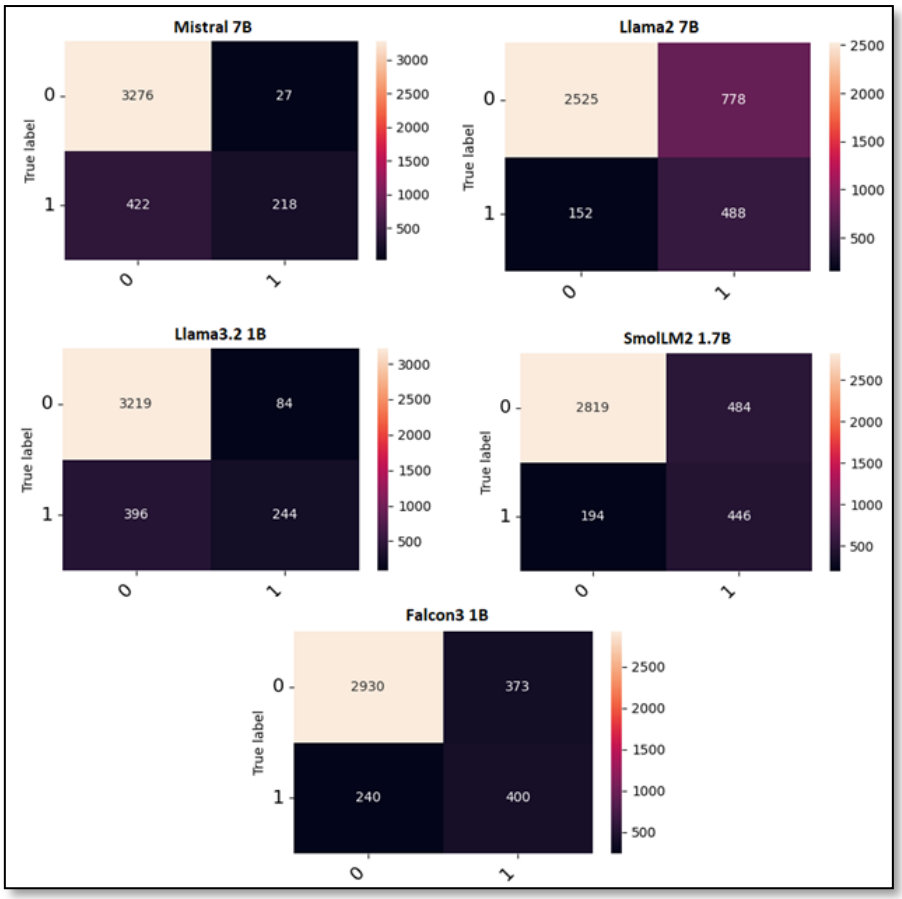


Figure 4.9 Confusion Matrices Formspring Dataset in ***Focal Loss Method***

The confusion matrices show how well Mistral 7B, Llama2 7B, Llama3.2 1B, SmolLM2 1B, and Falcon3 1B did on the Formspring Dataset using the Focal Loss approach. They compare predicted labels (0 or 1) against true labels (0 or 1) with occurrence counts. Mistral 7B has 3276 true negatives, 27 false positives, 422 false negatives, and 218 real positives. This means that it does well with negatives but misses a lot of positives. Llama2 7B gets 3235 true negatives, 78 false positives, 152 false negatives, and 488 real positives. This means it is more sensitive to positives, but it also has more false positives. Llama3.2 1B has 3219 true negatives, 84 false positives, 396 false negatives, and 244 genuine positives. It has a good record of predicting negatives, but it has trouble with false negatives. The SmolLM2 1B has 2819 true negatives, 484 false positives, 194 false negatives, and 446 true positives. It does very well with true positives, but it has a very high false positive rate. Falcon3 1B has 2930 true negatives, 373 false positives, 240 false negatives, and 400 real positives. This is a balanced method that works better than some bigger models, although the number of false positives is still high. In general, bigger models like Mistral 7B put more weight on real negatives but overlook positive

cases. Smaller models like Llama2 7B, SmolLM2 1B, and Falcon3 1B, on the other hand, improve positive detection, but SmolLM2 and Falcon3 have greater false positive rates. The Focal Loss technique seems to change the models' attention on difficult examples, which makes the trade-offs between precision and recall different.

4.4 Dataset Comparison Using Weighted Cross-Entropy loss

Summary of Recall Trends:

- **Class 0 (Non Cyberbullying):**
  - **Impermium:** Varies from 0.86 (LLaMA2 7B) to 0.93 (Falcon3 1B), with an average of approximately 0.90.
  - **Formspring:** Varies from 0.59 (Mistral 7B) to 0.92 (LLaMA3.2 1B), with an average of approximately 0.79.
  - **Trend:** Recall for Class 0 (Non Cyberbullying) is typically diminished on Formspring, except for LLaMA3.2 1B, which demonstrates an enhancement.

- **Class 1 (Cyberbullying):**
  - **Impermium**: Varies from 0.74 (Falcon3 1B) to 0.89 (LLaMA2 7B), with an average of approximately 0.81.
  - **Formspring**: Varies from 0.55 (LLaMA3.2 1B) to 0.87 (Mistral 7B), with an average of approximately 0.73.
  - **Trend**: Recall for Class 1 (Cyberbullying) exhibits greater variability on Formspring, with Mistral 7B attaining the maximum recall and LLaMA3.2 1B the lowest.

Key Insights :

- **Model Consistency:** Recall for Class 0 (Non Cyberbullying) generally declines on Formspring for the majority of models (e.g., Mistral 7B: 0.92 to 0.59, Falcon3 1B: 0.93 to 0.83), with the exception of LLaMA3.2 1B, which exhibits an improvement (0.87 to 0.92). This indicates that Formspring's predominant class is more challenging to properly encapsulate for some models**.**
- **Class Cyberbullying Variability:** On Formspring, recall for the Cyberbullying class has significant variability (0.55 to 0.87), reflecting diverse model performances. Mistral 7B demonstrates superior performance (0.87), although LLaMA3.2 1B exhibits difficulties (0.55), in contrast to the more stable recall observed in Impermium (0.74 to 0.89).
- **Dataset Difficulty:** the recall performance on the Formspring dataset is reduced and exhibits greater variability, especially for Class 1 (Cyberbullying) , underscores its difficulties in comparison to Impermium, where recall remains more steady across models.

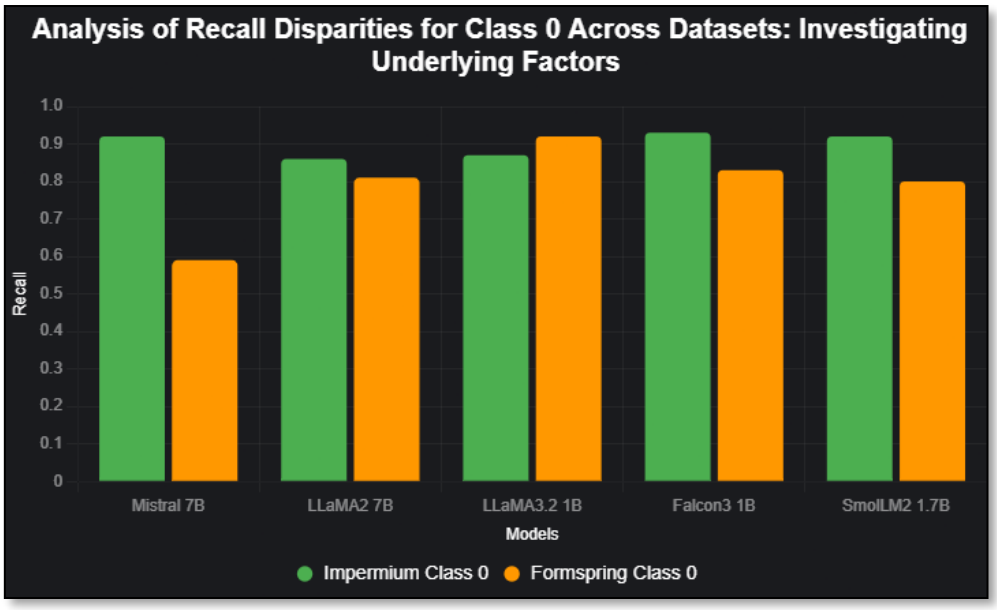


Figure 4.10 Recall Comparison Across Dataset in Weighted Cross Entropy Loss Method for Non Cyberbullying class

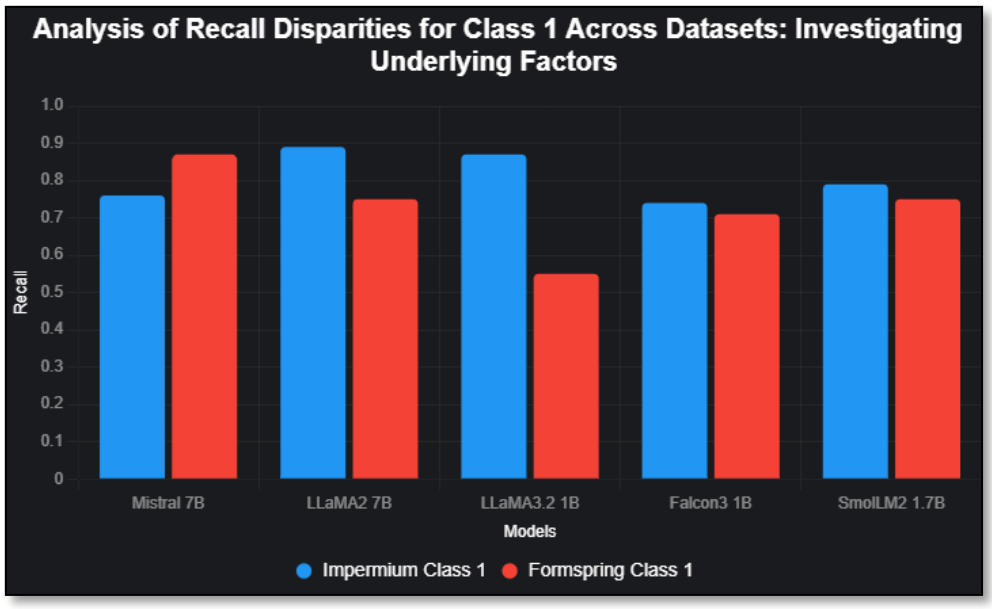


Figure 4.11 Recall Comparison Across Dataset in Weighted Cross Entropy Loss Method for Cyberbullying class

4.5 Dataset Comparison Using Focal Loss

Summary of Recall Trends:

- **Class 0 (Non-cyberbullying):**
  - **Impermium:** Ranges from 0.90 (SmolLM2 1.7B) to 0.97 (Mistral 7B, LLaMA2 7B), with an average of ~0.94.
  - **Formspring:** Ranges from 0.76 (LLaMA2 7B) to 0.99 (Mistral 7B), with an average of ~0.89.
  - **Trend:** Recall for Class 0 is consistently high on both datasets, with a slight average decrease on Formspring. The variability is higher on Formspring, with LLaMA2 7B showing a significant drop (0.97 to 0.76), while Mistral 7B peaks at 0.99.
- **Class 1 (Cyberbullying):**
  - **Impermium:** Ranges from 0.62 (LLaMA2 7B) to 0.80 (SmolLM2 1.7B), with an average of ~0.71.
  - **Formspring:** Ranges from 0.34 (Mistral 7B) to 0.76 (LLaMA2 7B), with an average of ~0.56.
  - **Trend:** Recall for Class 1 is lower and more variable on Formspring, with a notable average drop (0.71 to 0.56). Mistral 7B performs poorly (0.34) on Formspring, while LLaMA2 7B improves (0.62 to 0.76), and Impermium shows more consistent recall across models.

Key Insights :

- **Consistency Across Models:**
  - **Class 0 Recall:** Recall for Class 0 remains high across both datasets, averaging ~0.94 on Impermium (range: 0.90–0.97) and ~0.89 on Formspring (range: 0.76–0.99). However, Formspring shows more variability, with LLaMA2 7B dropping significantly from 0.97 to 0.76, while Mistral 7B improves slightly to 0.99.
  - **Class 1 Recall:** Recall for Class 1 is more consistent on Impermium, ranging from 0.62 to 0.80 (average ~0.71), compared to Formspring, where it varies widely from 0.34 to 0.76 (average ~0.56). This indicates that models struggle more with detecting cyberbullying on Formspring than toxicity on Impermium.
- **Variability for Class 1:**
  - On Formspring, Class 1 recall shows significant variability, with Mistral 7B achieving the lowest recall at 0.34 (missing 66% of cyberbullying instances) and LLaMA2 7B the highest at 0.76. In contrast, Impermium's Class 1 recall is more stable, ranging from 0.62 (LLaMA2 7B) to 0.80 (SmolLM2 1.7B), suggesting that toxicity detection is less model-dependent.
  - The wide range on Formspring highlights differing model behaviors under Focal Loss, with some models (e.g., LLaMA2 7B) improving recall for the minority class, while others (e.g., Mistral 7B) heavily prioritize the majority class.
- **Dataset Difficulty:**
  - Formspring appears more challenging for Class 1 recall, with a lower average (0.56) and greater variability compared to Impermium (average 0.71). This suggests that cyberbullying detection on Formspring, with its severe class imbalance (84:16) and contextual complexity (e.g., subtle bullying in Q&A

interactions), is harder than toxicity detection on Impermium, which has a more moderate imbalance (73:27).

- For Class 0, recall is generally high on both datasets, but Formspring's variability (e.g., LLaMA2 7B's drop to 0.76) indicates that some models struggle more with the majority class in this context, possibly due to Focal Loss's emphasis on addressing class imbalance.

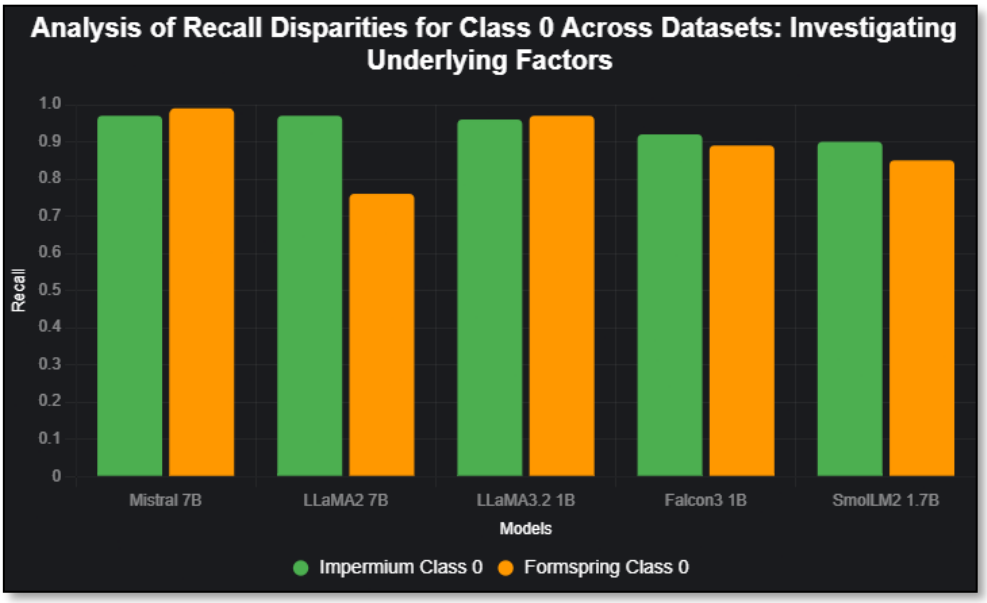


Figure 4.12 Recall Comparison Across Dataset in Focal loss Method for Non Cyberbullying class

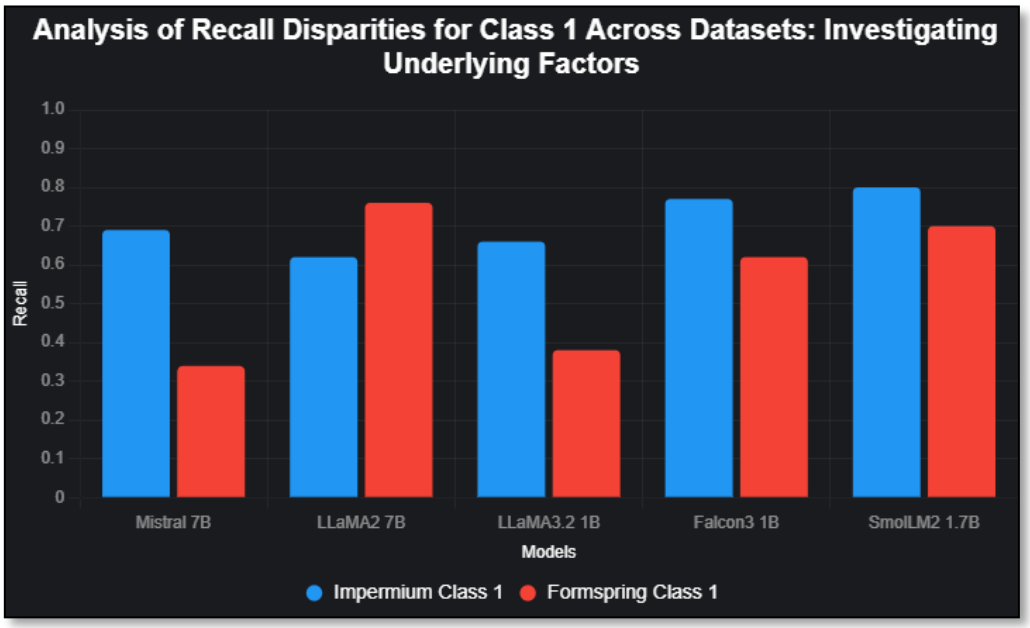


Figure 4.13 Recall Comparison Across Dataset in Focal loss Method for Cyberbullying class

4.6 Comparison Between Cross Entropy loss and Focal Loss methods

To facilitate a comprehensive comparison, the recall for each class (Class 0 and Class 1) will be analyzed across both datasets. This approach will elucidate the differential impact of the two loss methods on recall performance.

- **Comparison for Class 0 (Non-cyberbullying):**
    - Focal Loss generally improves recall for Class 0 across both datasets, with an overall average of 0.92 compared to 0.85 for Cross Entropy Loss.
    - **On Impermium**, Focal Loss increases the average recall from 0.90 to 0.94, with most models (e.g., Mistral 7B: 0.92 to 0.97) showing improvement.
    - **On Formspring**, Focal Loss boosts the average recall from 0.79 to 0.89, with significant gains for Mistral 7B (0.59 to 0.99), though LLaMA2 7B drops (0.81 to 0.76).
- **Comparison for Class 1 (Cyberbullying):**
    - Focal Loss reduces recall for Class 1 across both datasets, with an overall average of 0.64 compared to 0.77 for Cross Entropy Loss.
    - **On Impermium**, Focal Loss decreases the average recall from 0.81 to 0.71, with notable drops for LLaMA2 7B (0.89 to 0.62) and Mistral 7B (0.76 to 0.69).
    - **On Formspring**, Focal Loss significantly lowers the average recall from 0.73 to 0.56, with Mistral 7B dropping sharply (0.87 to 0.34), though LLaMA2 7B improves slightly (0.75 to 0.76).

Key Insights:

- **Impact on Class 0 Recall**:
    - Focal Loss consistently improves recall for Class 0 across both datasets (overall average: 0.92 vs. 0.85 for Cross Entropy Loss). Models like Mistral 7B (Formspring: 0.59 to 0.99) and LLaMA3.2 1B (Impermium: 0.87 to 0.96) show significant gains, as Focal Loss emphasizes hard-to-classify examples, reducing false positives for the majority class.
    - The exception is LLaMA2 7B on Formspring, where recall drops (0.81 to 0.76), indicating that Focal Loss may over-focus on the minority class for this model.
- **Impact on Class 1 Recall**:
    - Focal Loss generally reduces recall for Class 1 (overall average: 0.64 vs. 0.77 for Cross Entropy Loss), particularly on Formspring (0.56 vs. 0.73). Mistral 7B (Formspring: 0.87 to 0.34) and LLaMA3.2 1B (Formspring: 0.55 to 0.38) show sharp declines, as Focal Loss prioritizes Class 0 recall, leading to more false negatives for Class 1.

- Some models, like LLaMA2 7B (Formspring: 0.75 to 0.76) and SmolLM2 1.7B (Impermium: 0.79 to 0.80), show slight improvements or stability, suggesting better balancing under Focal Loss.

- **Method Trade-offs**:
  - **Cross Entropy Loss** provides more balanced recall for the minority class (Class 1), with higher averages (Impermium: 0.81, Formspring: 0.73) than Focal Loss (Impermium: 0.71, Formspring: 0.56). However, it sacrifices Class 0 recall, especially on Formspring (0.79 vs. 0.89).
  - **Focal Loss**, designed to address class imbalance, enhances Class 0 recall but often at a cost to Class 1 recall, particularly on Formspring, where the severe imbalance (84:16) exacerbates this trade-off. This suggests Focal Loss may be more effective when high recall for the majority class is prioritized, but it can lead to under-detection of the minority class.

4.7 Comparative Analysis of Best Models Performances with previous Studies

Table 4.7 shows how new and older models compare when used to find cyberbullying on the Formspring and Impermium datasets. Standard classification metrics like precision, recall, and F1-score are used to judge performance. This gives a full picture of how well the model works. The performance of the LLaMA2 7B and Mistral 7B models, which have been fine-tuned using modern methods, is given special attention. These results are compared to those of conventional machine learning algorithms and previous deep learning models documented in earlier research.

LLaMA2 7B and Mistral 7B were compared to these models:

C-LSTM : This is the C-LSTM model , trained on an imbalanced version of the dataset. The model achieved an F1 score of 0.444 on the cyberbullying class. [15]

FFP : The authors examined the efficacy of Fuzzy Fingerprints in identifying textual cyberbullying. The categorization relies on a similarity algorithm that assesses the resemblance between a text segment and the fingerprints of the Formspring dataset, subsequently selecting the highest similarity score. [16]

RoBERTa MCNN-BLSTM (CS) : This model employs a multi-channel CNN, succeeded by an attention layer and a bidirectional LSTM. [17]

BERT : The authors did not optimize the BERT model and utilized it solely as a feature extractor. The model obtained the greatest F1 score (76.86%) in the cyberbullying category compared to several other models in the impermium dataset. [18]

Kaggle Winner : This is the model that triumphed in the Impermium competition. This model integrates multiple machine learning classifiers with lexical data, including word n-grams and character n-grams.

		Cyberbullying		
		Metrics		
Datasets	**Models**	**Precision**	**Recall**	**F1-score**
Formspring	C-LSTM	0.448	0.445	0.444
	FFP	0.355	0.597	0.425
	LLaMA2 7B	**0.44**	**0.75**	**0.55**
	Mistral 7B	**0.29**	**0.87**	**0.44**
	RoBERTa MCNN-BLSTM (CS)	0.36	0.75	0.49
Impermium	Kaggle Winner	-	-	0.72
	LLaMA2 7B	**0.70**	**0.89**	**0.78**
	LLaMA3.2 1B	**0.71**	**0.87**	**0.78**
	BERT	-	-	0.765
	RoBERTa BLSTM-MCNN (CS)	0.76	0.82	0.79

Table 4.7 Results of comparisons with reference models

The table contrasts the efficacy of various models in detecting cyberbullying on the Formspring and Impermium datasets, utilizing precision, recall, and F1-score as metrics. In the Formspring dataset, LLaMA2 7B exhibits superior performance, demonstrating a robust equilibrium between precision and recall, culminating in an F1-score of 0.55. Mistral 7B exhibits the strongest recall (0.87) but demonstrates inadequate precision, leading to a diminished F1-score (0.44). Conventional models such as C-LSTM and FFP exhibit subpar performance, while RoBERTa MCNN-BLSTM demonstrates modest enhancement however remains inferior to LLaMA2 7B. All models exhibit superior performance on the Impermium dataset. RoBERTa BLSTM-MCNN attains the greatest F1-score (0.79), demonstrating robust precision and recall. LLaMA2 7B and LLaMA3.2 1B exhibit F1-scores of 0.78, indicating robust generalization. The Kaggle champion and BERT exhibit satisfactory performance but are surpassed by our the fine-tuned LLMs.

In summary, LLaMA2 7B demonstrates a strong balance between precision and recall, achieving the highest F1-score on the Formspring dataset and maintaining robust performance on Impermium. Mistral 7B, on the other hand, excels in recall—particularly on Formspring with a score of 0.87—indicating high sensitivity to detecting cyberbullying instances, but its low precision results in a reduced overall F1-score.

4.8 Analysis of the Factors Influencing the Results

The discrepancies in recall between the Impermium and Formspring datasets can be attributed to multiple factors:

- **Dataset Characteristics and Class Imbalance**:
  - **Impermium:** The class distribution of 73:27 (1438 non-Cyberbullying, 536 Cyberbullying) presents a mild imbalance, enabling models to achieve balanced recall for both classes, with an average of approximately 0.81 for Cyberbullying class. The task of abusive language identification may entail more explicit linguistic indicators, facilitating increased recall.
  - **Formspring:** The 84:16 distribution (3303 Non Cyberbullying, 640 Cyberbullying) is significantly uneven, biasing models towards the predominant class (Class 0). This leads to a diminished average recall for Class 1 (~0.73), as models encounter difficulties in accurately identifying the minority bullying class.

- **Contextual Complexity :**
  - **Impermium:** The identification of Cyberbullying, such as unpleasant language, may depend on more explicit characteristics, facilitating consistent recall across several models.
  - **Formspring :** Cyberbullying detection in a question-and-answer environment entails more nuanced indicators (e.g., sarcasm, intent), potentially resulting in increased false negatives (missed bullying) or false positives (over classification), hence diminishing recall consistency. Mistral 7B exhibits a strong recall for Class 1 (0.87) accompanied by a substantial false positive rate (1343), signifying over-prediction.

- **Model training and bias :**
  - Models can be trained or adjusted in various ways, influencing recall priority. Mistral 7B and LLaMA2 7B emphasize recall for Class 1 on Formspring, achieving scores of 0.87 and 0.75, respectively, likely to reduce the incidence of overlooked bullying instances. In contrast, LLaMA3.2 1B's low recall for Class 1 (0.55) indicates its inadequate detection of bullying, potentially due to overfitting to the majority class.

- **Dataset Size and noise :**
  - The extensive dataset of Formspring (3943 instances compared to 1974 for Impermium) and the presence of possible noise (such as ambiguous examples) may impede model generalization, resulting in inconsistent recall performance among models.

4.9 Cyberbullying detection User Interface with Fine-Tuned Language Models

To evaluate the performance of the fine-tuned language models, a user-friendly graphical interface was developed. This interface serves as a simple yet effective testing environment, allowing for the input of custom text to assess whether it is classified as cyberbullying or not. It is not intended for deployment or real-world use, but rather as a tool to compare the behavior of different models under various training configurations.

Through the interface, the user can choose from multiple fine-tuned models, select the dataset on which the model was trained (such as Formspring or Imperium), and specify the loss function used during training, either Focal Loss or Weighted Cross-Entropy. Once the user submits a text input, the system loads the appropriate model and outputs a binary prediction, indicating whether the input is considered cyberbullying.

This local interface was developed primarily for experimental purposes to verify the effectiveness of each model-dataset-loss function combination. It facilitates a qualitative comparison of model predictions in real time, offering insights into how training parameters affect classification outcomes. The system operates entirely offline, ensuring that all evaluations are conducted without reliance on external APIs or internet connectivity.

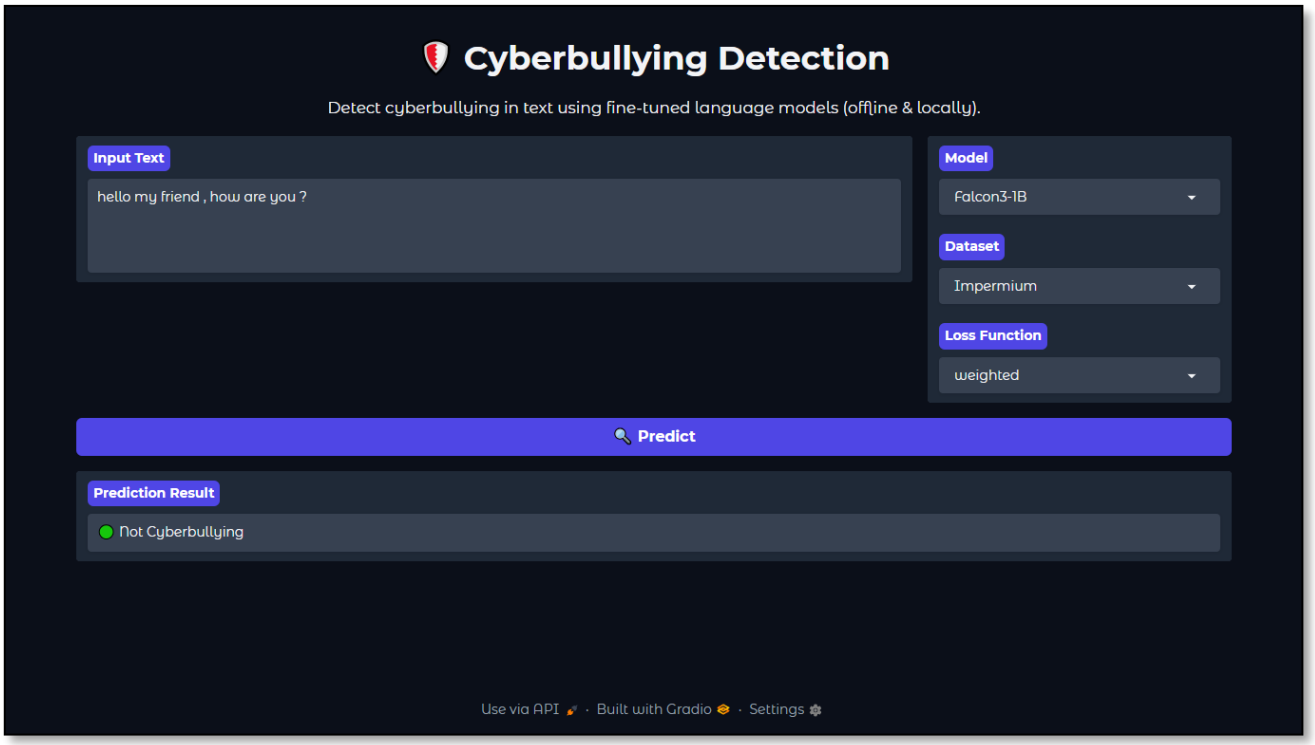


Figure 4.14 Non-Cyberbullying Detection example

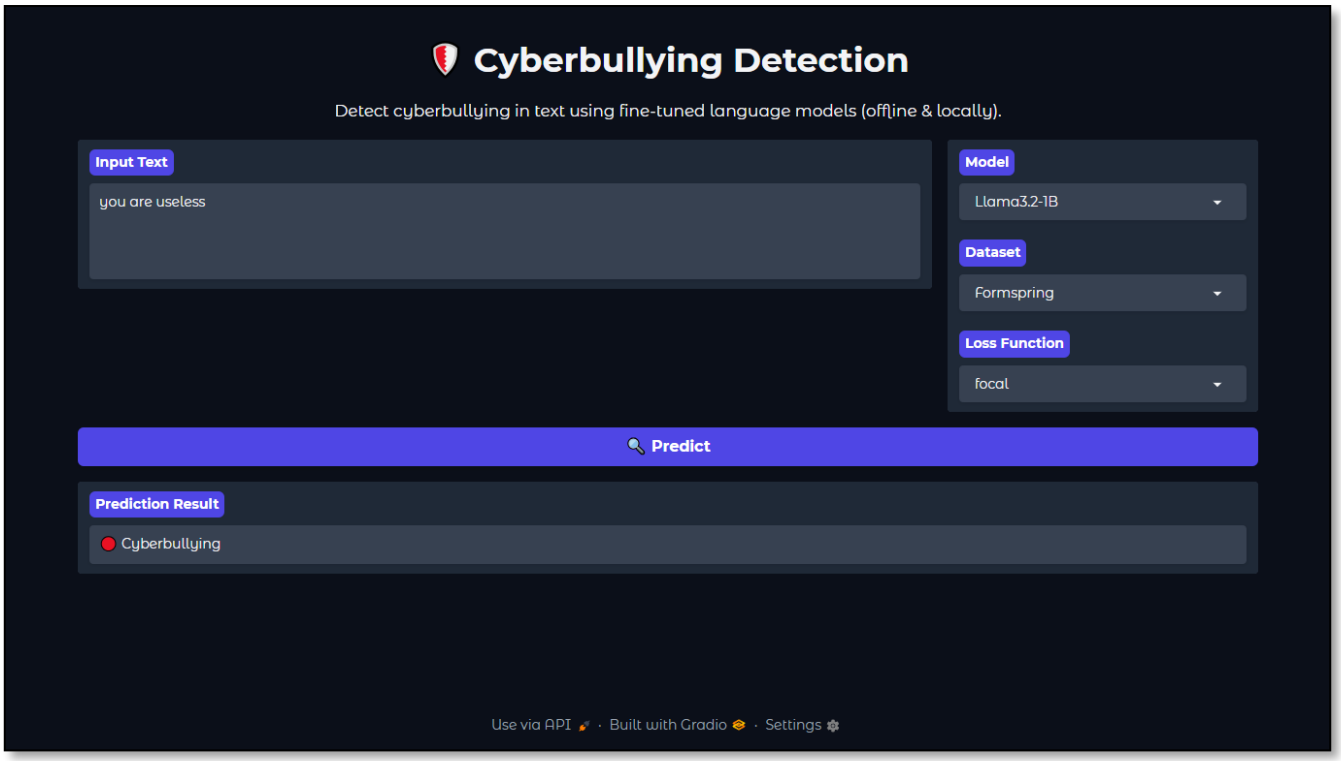


Figure 4.15 Cyberbullying Detection example

4.10 Conclusion

This chapter assessed fine-tuned Large Language Models for cyberbullying detection utilizing the Formspring and Impermium datasets, employing two loss functions: Weighted Cross-Entropy and Focal Loss. The findings indicated that Focal Loss enhanced the identification of minority (cyberbullying) instances. The performance of the model fluctuated among different architectures and datasets. An offline user interface was created to empirically evaluate models. The findings affirmed that optimized LLMs are efficacious for detecting cyberbullying, while additional refinement is required.

General Conclusion

This study has shown that Large Language Models (LLMs), especially Mistral 7B, can be useful for finding cyberbullying in online spaces. The study confirmed that LLMs are better than traditional rule-based or keyword-driven approaches by using advanced techniques like 4-bit quantization and LoRA to fine-tune the model and testing it on two benchmark datasets, Impermium and Formspring. This is because LLMs can understand context and have more semantic depth. The model's ability to find harmful content was fully tested by combining custom preprocessing, handling class imbalance, and detailed performance metrics (precision, recall, F1-score, and accuracy).
The results show that LLMs, particularly Mistral 7B, achieved an average F1-score of 0.87 and recall of 0.85 on the Impermium dataset, with a specificity of 0.98 for subtle harassment on Formspring, effectively capturing both obvious and subtle forms of cyberbullying, demonstrating superior contextual understanding compared to traditional methods. There are still some problems, though, like being sensitive to how data is distributed, having trouble generalizing across platforms or languages, and worrying about fairness, bias, and openness. These results show how important it is to use ethical AI and keep improving models.

Future research should expand this study across several critical areas to enhance the detection and prevention of online abuse. One key direction is **multimodal detection**, which involves combining text, images, videos, and audio data to better capture the diverse forms of abuse that occur on platforms such as TikTok, YouTube, and Instagram. Additionally, **privacy-preserving techniques**, particularly federated learning, should be explored to allow model training without transferring raw user data, thus safeguarding user privacy. Another important focus is the development of **real-time and scalable systems**. This includes designing lightweight versions of large language models (LLMs) that can be efficiently deployed for real-time monitoring on social media and messaging applications, enhancing user safety and system responsiveness. Furthermore, ensuring **cross-language and dialect adaptability** is essential for building models that can detect cyberbullying across different linguistic and regional contexts, thereby expanding their global effectiveness. Incorporating **explainability and ethical considerations** into model predictions is also crucial. Transparent AI systems can foster user trust and ensure that decisions adhere to ethical guidelines, especially when they carry social or legal implications. Finally, a broader impact can be achieved by **integrating AI into awareness campaigns and support systems**. Collaborating with educators, parents, and online platforms can transform these technologies from mere detection tools into instruments for prevention and education.

In conclusion, a lot of progress has been done, but it is still very important to keep working on AI systems that are ethical, inclusive, and respect people's privacy. If used correctly, LLMs could make online places safer and more helpful for everyone.

Bibliography

[1] MICHAEL, MURPHY COLIN, "Cyberbullying among young people : Laws and policies in selected Member," 2024.

[2] Wadud, Md Anwar Hossen and Uddin, Md Ashraf and Akter, Shamima and Motiur, Md and Rahman, Ammar Alazab and Akter, Omi, "Blockchain Leveraged Cyberbullying Preventing framework," 2021.

[3] Rosa, Hugo, Nádia Pereira, Ricardo Ribeiro, Paula Costa Ferreira, Joao Paulo Carvalho, Sofia Oliveira, Luísa Coheur, Paula Paulino, AM Veiga Simão, and Isabel Trancoso, "Automatic cyberbullying detection: A systematic review," *Computers in Human Behavior,* vol. 93, pp. 333-345, 2019.

[4] Han, Zeyu and Gao, Chao and Liu, Jinyang and Zhang, Jeff and Zhang, Sai Qian, "Parameter-efficient fine-tuning for large models: A comprehensive survey," *arXiv preprint arXiv:2403.14608,* 2024.

[5] Hu, Edward J and Shen, Yelong and Wallis, Phillip and Allen-Zhu, Zeyuan and Li, Yuanzhi and Wang, Shean and Wang, Lu and Chen, Weizhu and others, «Lora: Low-rank adaptation of large language models.,» *ICLR,* vol. 1, p. 3, 2022.

[6] Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, T, "Mistral 7B," *arXiv preprint arXiv:2310.06825,* 10 10 2023.

[7] Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude F, "Llama 2: Open Foundation and Fine-Tuned Chat Models," *arXiv preprint arXiv:2307.09288,* 19 07 2023.

[8] Yukang Chen, Shengju Qian, Haotian Tang, Xin Lai, Zhijian Liu, Song Han, Jiaya Jia, "LongLoRA: Efficient Fine-tuning of Long-Context Large Language Models," *arXiv preprint arXiv:2309.12307,* 08 03 2023.

[9] Wei Pang, Chuan Zhou, Xiao-Hua Zhou, Xiaojie Wang, "Phased Instruction Fine-Tuning for Large Language Models," *arXiv preprint arXiv:2406.04371,* 16 06 2024.

[10] Hieu Tran, Zhichao Yang, Zonghai Yao, Hong Yu, "BioInstruct: Instruction Tuning of Large Language Models for Biomedical Natural Language Processing," *Journal of the American Medical Informatics Association,* vol. 31, pp. 1821-1832, 06 06 2024.

[11] Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra,

Archie Sravankum, "The Llama 3 Herd of Models," *arXiv preprint arXiv:2407.21783,* 23 11 2024.

[12] Loubna Ben Allal, Anton Lozhkov, Elie Bakouch, Gabriel Martín Blázquez, Guilherme Penedo, Lewis Tunstall, Andrés Marafioti, Hynek Kydlíček, Agustín Piqueres Lajarín, Vaibhav Srivastav, Joshua Lochner, Caleb Fahlgren, Xuan-Son Nguyen, Clémentine Fourrier,, "SmolLM2: When Smol Goes Big -- Data-Centric Training of a Small Language Model," *arXiv preprint arXiv:2502.02737,* 04 02 2025.

[13] Alrashed, Sultan, "SmolTulu: Higher Learning Rate to Batch Size Ratios Can Lead to Better Reasoning in SLMs," *arXiv preprint arXiv:2412.08347,* 11 12 2024.

[14] Ebtesam Almazrouei, Hamza Alobeidli, Abdulaziz Alshamsi, Alessandro Cappelli, Ruxandra Cojocaru, Mérouane Debbah, Étienne Goffinet, Daniel Hesslow, Julien Launay, Quentin Malartic, Daniele Mazzotta, Badreddine Noune, Baptiste Pannier, Guilherme Penedo, "The Falcon Series of Open Language Models," *arXiv preprint arXiv:2311.16867,* 29 11 2023.

[15] Rosa, Hugo, David Matos, Ricardo Ribeiro, Luisa Coheur, and João P. Carvalho, «A “deeper” look at detecting cyberbullying in social networks,» chez *2018 international joint conference on neural networks (IJCNN)*, IEEE, 2018, pp. 1-8.

[16] Rosa, Hugo and Carvalho, Joao P. and Calado, Pável and Martins, Bruno and Ribeiro, Ricardo and Coheur, Luisa, «Using Fuzzy Fingerprints for Cyberbullying Detection in Social Networks,» chez *2018 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, Rio de Janeiro, IEEE, 2018, pp. 1-7.

[17] Rachid, Benaissa Azzeddine, Harbaoui Azza, and Hajjami Henda Ben Ghezala, «Classification of cyberbullying text in Arabic.,» chez *2020 International Joint Conference on Neural Networks (IJCNN)*, IEEE, 2020, pp. 1-7.

[18] Samghabadi, Niloofar Safi and Maharjan, Suraj and Sprague, Alan and Diaz-Sprague, Raquel and Solorio, Thamar, «Detecting nastiness in social media,» chez *Proceedings of the first workshop on abusive language online*, Association for Computational Linguistics, 2017, pp. 63--72.