

TECHNICAL CERTIFICATE



Certificate of Completion

Awarded to

Samanvitha Bolem

Upon successfully Completed the Boot Camp Training in Machine Learning
for 45 Hrs from 01st Sep 2021 To 27th Sep 2021



Nikhil Barshikar
Director



Certificate of Completion

Awarded to

Shibani P

Upon successfully Completed the Boot Camp Training in Machine Learning
for 45 Hrs from 01st Sep 2021 To 27th Sep 2021



A handwritten signature in black ink, appearing to read 'Nikhil', located in the bottom right corner of the certificate.

Nikhil Barshikar
Director

ABSTRACT

The world has entered an era of social media. Digital media and apps allow children to communicate and express their creativity, connect with peers, and share their feelings. However, they can be an avenue through which [cyberbullying](#) occurs. There are many types of apps and sites available for free that give users the ability to search for people and share or post information about them anonymously, with the prevalence of social media and digital forums, comments, photos, posts, and content shared by individuals can often be viewed by strangers as well as acquaintances. Cyberbullying Detection implements our algorithms, in finding a negative comment from the messages it receives by a user. The algorithm gives a value to the message based on our pre-trained data and then decides if the text is rude to be brought into notice.

TABLE OF CONTENTS

CHAPTER NO.	TITLE	PAGE NUMBER
1	INTRODUCTION	
1.1	Aim and scope of investigation	1
1.2	Skills required to thrive in this field	1
1.3	Python in Software	2
1.4	Strengths and Weakness of Python	2
1.5	Installing Python	3
2	EXPERIMENTAL MATERIALS OR METHOD ALGORITHMS USED	
2.1	Sentiment Analysis	5
2.2	Approaches for Sentiment Analysis	6
2.3	Classification of Techniques	7
2.4	Limitations	8

3	BERT	
3.1	Introducing BERT	9
3.2	Pretraining BERT	10
3.3	Fine Tuning BERT	11
3.4		
4	RESULTS AND DISCUSSION, PERFORMANCE ANALYSIS	
4.1	Result	13
4.2	Analysis	14
4.3	Discussions	
4.4		
5	SUMMARY & CONCLUSION	19
5.1	Summary	19
	5.1.1 Browser configuration	
	5.1.2 Operating system	

5.2	Conclusion	19
6	SUPPORT	
6.1	Hardware	
6.2	Software	
6.3	SRS	
6.4	Non-functional Requirements	
	REFERENCES	20
	APPENDIX	21
	A) SOURCE CODE	

LIST OF FIGURES

FIGURE NUMBER	TITLE	PAGE NUMBE R
1	Flow Chart	5
2	System Architecture	9
3	Bert Representation	12
4	Bert Result	13
5	Result Values	14
6	Data Distribution	15
7	Screenshot	15
8	Detection 1	16
9	Detection 2	16

CHAPTER 1

INTRODUCTION

1.1 AIM AND SCOPE OF THE INVESTIGATION:

Utilization of web and online media foundations tends in the utilization of sending, getting what's more posting of negative, hurtful, bogus, or mean substance about another person which accordingly implies Cyberbullying? Harassing over web-based media additionally works equivalent to undermining, slander, what's more berating the person. Cyberbullying has prompted a serious expansion in psychological well-being issues, particularly among the youthful age. It has brought about lower confidence, expanded self-destructive ideation. Except if some action against cyberbullying is taken, confidence what's more psychological well-being issues will influence a whole age of youthful grown-ups. A considerable lot of the customary AI models have been carried out in the past for the programmed discovery of cyberbullying via web-based media. In any case, these models have not thought about every one of the fundamental elements that can be utilized to distinguish or order an assertion or post as tormenting. In this paper, we proposed a model dependent on different highlights that ought to be thought of while distinguishing cyberbullying and execute a couple of elements with the assistance of a bidirectional profound learning model called BERT.

1.2 SKILLS REQUIRED TO THRIVE IN THIS FIELD

Creativity

Imagination

Ethics

Social Morality

Good computer knowledge

Knowledge about coding languages

Knowledge about relevant software

1.3 PYTHON IN SOFTWARE

Python is a popular object-oriented programming language having the capabilities of high-level programming language. Its easy to learn syntax and portability capability makes it popular these days. The followings facts gives us the introduction to Python –

- Python was developed by Guido van Rossum at Stichting Mathematisch Centrum in the Netherlands.
- It was written as the successor of programming language named 'ABC'.
- It's first version was released in 1991.
- The name Python was picked by Guido van Rossum from a TV show named Monty Python's Flying Circus.
- It is an open source programming language which means that we can freely download it and use it to develop programs. It can be downloaded from www.python.org.
- Python programming language is having the features of Java and C both. It is having the elegant 'C' code and on the other hand, it is having classes and objects like Java for object-oriented programming.
- It is an interpreted language, which means the source code of Python program would be first converted into bytecode and then executed by Python virtual machine.

1.4 STRENGTHS AND WEAKNESS OF PYTHON

Every programming language has some strengths as well as weaknesses, so does Python too.

STRENGTHS

According to studies and surveys, Python is the fifth most important language as well as the most popular language for machine learning and data science. It is because of the following strengths that Python has –

Easy to learn and understand – The syntax of Python is simpler; hence it is relatively easy, even for beginners also, to learn and understand the language.

Multi-purpose language – Python is a multi-purpose programming language because it supports structured programming, object-oriented programming as well as functional programming.

Huge number of modules – Python has huge number of modules for covering every aspect of programming. These modules are easily available for use hence making Python an extensible language.

Support of open source community – As being open source programming language, Python is supported by a very large developer community. Due to this, the bugs are easily fixed by the Python community. This characteristic makes Python very robust and adaptive.

Scalability – Python is a scalable programming language because it provides an improved structure for supporting large programs than shell-scripts.

WEAKNESS

Although Python is a popular and powerful programming language, it has its own weakness of slow execution speed.

The execution speed of Python is slow as compared to compiled languages because Python is an interpreted language. This can be the major area of improvement for Python community.

1.5 INSTALLING PYTHON

For working in Python, we must first have to install it. You can perform the installation of Python in any of the following two ways –

- Installing Python individually
- Using Pre-packaged Python distribution – Anaconda

Let us discuss these each in detail.

Installing Python Individually

If you want to install Python on your computer, then then you need to download only the binary code applicable for your platform. Python distribution is available for Windows, Linux and Mac platforms.

The following is a quick overview of installing Python on the above-mentioned platforms –

On Unix and Linux platform

With the help of following steps, we can install Python on Unix and Linux platform –

- First, go to www.python.org/downloads/.
- Next, click on the link to download zipped source code available for Unix/Linux.
- Now, Download and extract files.

CHAPTER 3

3.1 BERT:

We introduce a new language representation model called BERT, which stands for Bidirectional Encoder Representations from Transformers. Unlike recent language representation models (Peters et al., 2018a; Radford et al., 2018), BERT is designed to pretrain deep bidirectional representations from unlabeled text by jointly conditioning on both left and right context in all layers. As a result, the pre-trained BERT model can be fine-tuned with just one additional output layer to create state-of-the-art models for a wide range of tasks, such as question answering and language inference, without substantial task specific architecture modifications.

We introduce BERT and its detailed implementation in this section. There are two steps in our framework: pre-training and fine-tuning. During pre-training, the model is trained on unlabeled data over different pre-training tasks. For finetuning, the BERT model is first initialized with the pre-trained parameters, and all the parameters are fine-tuned using labeled data from the downstream tasks. Each downstream task has separate fine-tuned models, even though they are initialized with the same pre-trained parameters. The question-answering example in Figure 1 will serve as a running example for this section. A distinctive feature of BERT is its unified architecture across different tasks. There is minimal difference between the pre-trained architecture and the final downstream architecture.