

# SARCASM DETECTION using deep learning

Hareetima Sonkar (Group-7)





# Problem Description

- Sarcasm is a complex sentiment expressed using intensified positive or positive words typically intended to convey a negative connotation.
- The challenge lies in interpreting sarcastic content correctly, especially in Natural Language Processing (NLP) and sentiment analysis.
- Our goal is to build a model that can recognize and understand sarcastic behaviour and patterns in text

# 1. Business Problem

**Misinterpreting sarcasm** can lead to inaccurate sentiment analysis, which can adversely affect businesses in various ways:

- Customer Feedback
- Public Relations
- Brand Reputation Management
- Social Media Monitoring
- Ad Campaign Analysis

## 2. Solution Description

### 1. Data Collection, Preprocessing & Cleaning:

- Data Loading
- Duplicate Removal, Removing null values etc.
- Data Summary
- removing punctuations, stopwords, etc.
- Visualizations
- Descriptive Statistics

### 2. Text Tokenization and Padding

- TensorFlow's Keras
- Padding type = 'post'

### 3. Model Training

- Model Architecture
- Data Splitting
- Training

### 4. Model Evaluation

- Performance Metrics
- Visualizations

### 5. Prediction

- The dataset contains text comments labelled as sarcastic (1) or non-sarcastic (0).
- Each comment is a short piece of text sourced from social media or other platforms [Reddit].
- **Dataset size: 1million (approx.)**

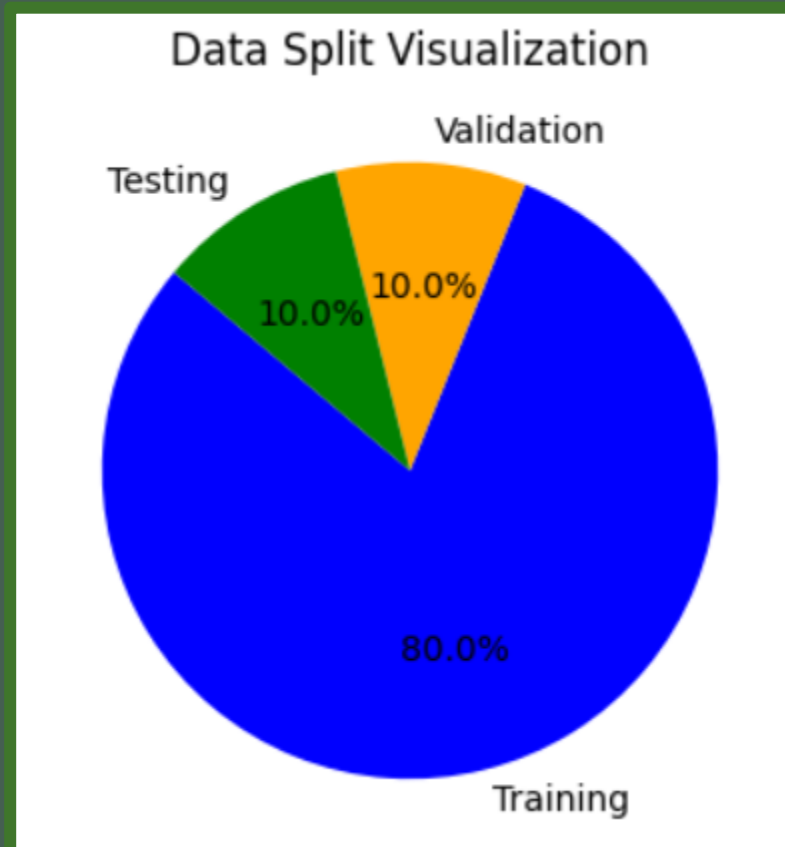
Examples:

0	LOOOOOL brrrxxxxton						
0	Ahh did not know that I rarely use that playstyle so I was just going off spellbuff alone						
0	Hes becoming a mene like Dicks out for Harambe						
1	If it wasnt for the invention of the coin flip Avilo would never lose						
0	I laughed so hard at how fucking stupid that was						
1	If the officer carried an extra gun this wouldnt have happened						
1	If it doesnt do well in Japan it doesnt matter anywhere else						
0	miere si rahat						
1	Becuase Hillary is a beacon of honesty						
1	Just wait for wretched eye with void burn that will be fun						
0	You can with adequate regulation						

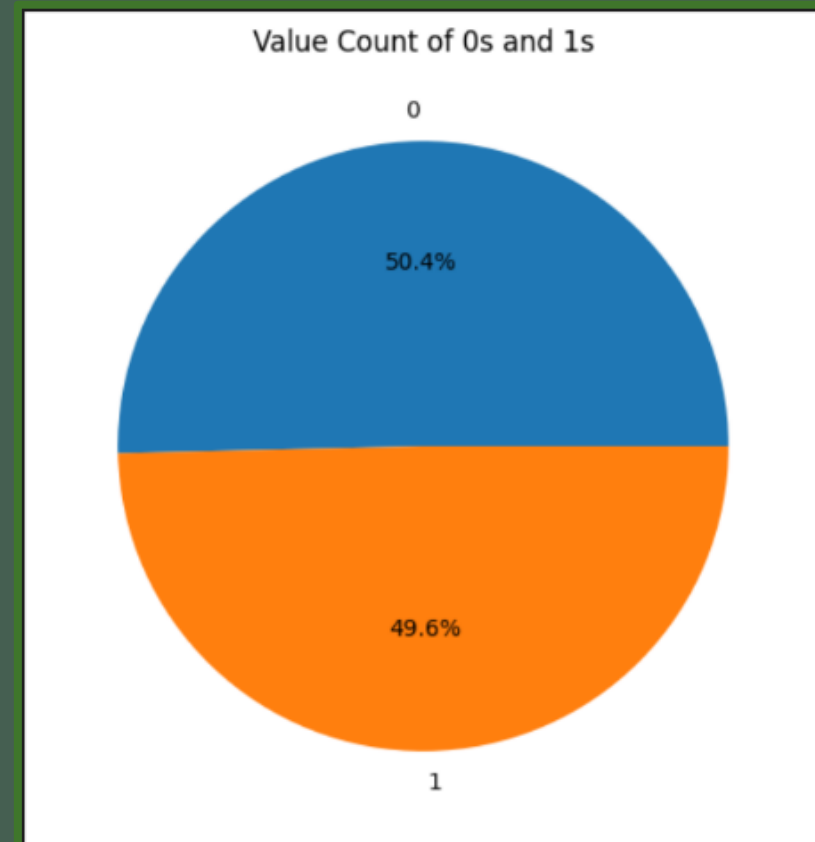
## 3. Dataset Description

## 4. Data Visualization

- Dataset Split:



- Label-wise Split:



## 5. Data Preprocessing

Text to lowercase; removing punctuations, duplicates, null values; new column for cleaned text; etc.

## 6. Tokenization and Embedding Techniques

- Tokenization: The text data is tokenized using the **Keras Tokenizer**.
- Embedding: The model uses an **Embedding layer** to convert tokens into dense vectors

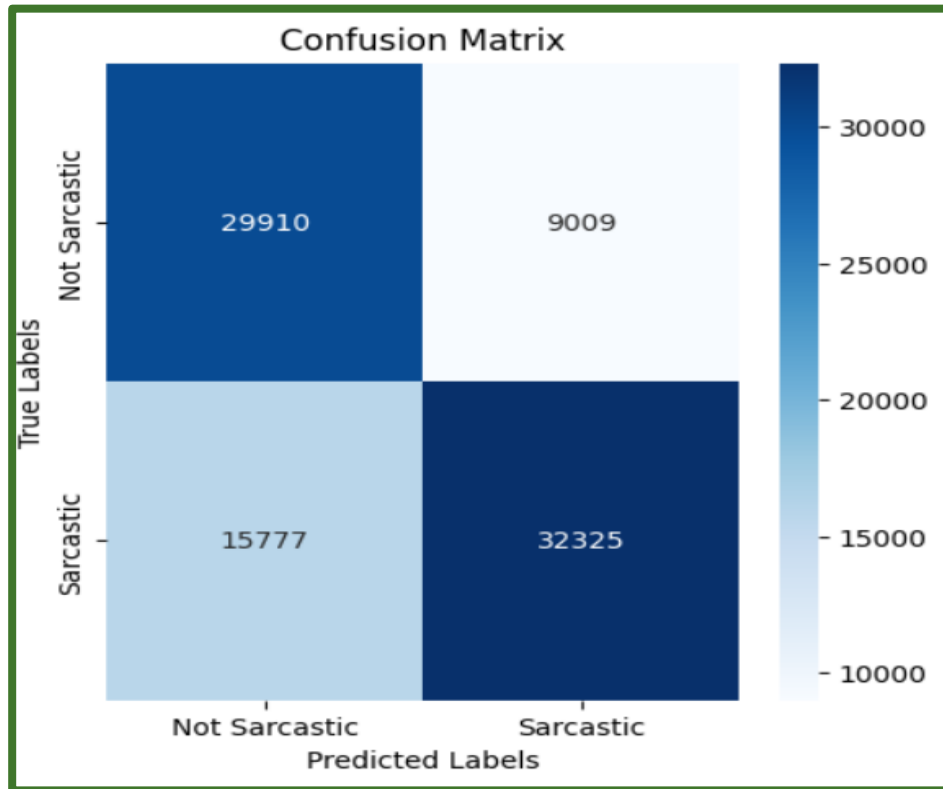
Model: "sequential"

Layer (type)	Output Shape	Param #
embedding (Embedding)	(None, 60, 200)	2,000,000
global_max_pooling1d (GlobalMaxPooling1D)	(None, 200)	0
dense (Dense)	(None, 40)	8,040
dropout (Dropout)	(None, 40)	0
dense_1 (Dense)	(None, 20)	820
dropout_1 (Dropout)	(None, 20)	0
dense_2 (Dense)	(None, 10)	210
dropout_2 (Dropout)	(None, 10)	0
dense_3 (Dense)	(None, 1)	11

## 7. Modelling



## Confusion Matrix:



## Classification Report:

Classification Report:				
	precision	recall	f1-score	support
Not Sarcastic	0.65	0.77	0.71	38919
Sarcastic	0.78	0.67	0.72	48102
accuracy			0.72	87021
macro avg	0.72	0.72	0.71	87021
weighted avg	0.73	0.72	0.72	87021



**THANK YOU**