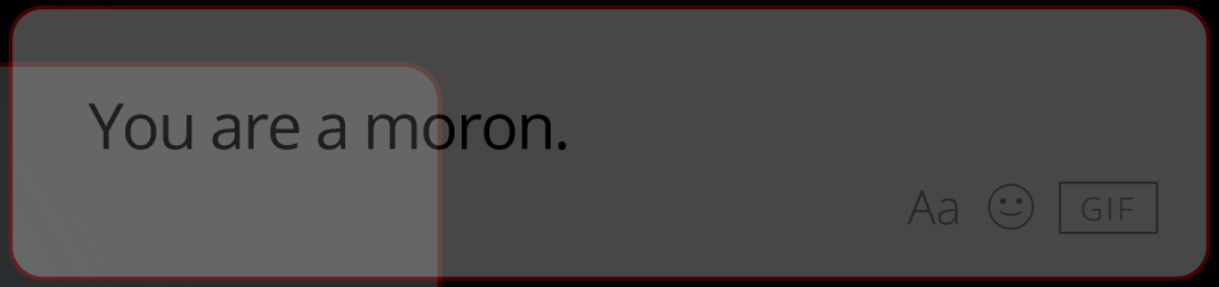
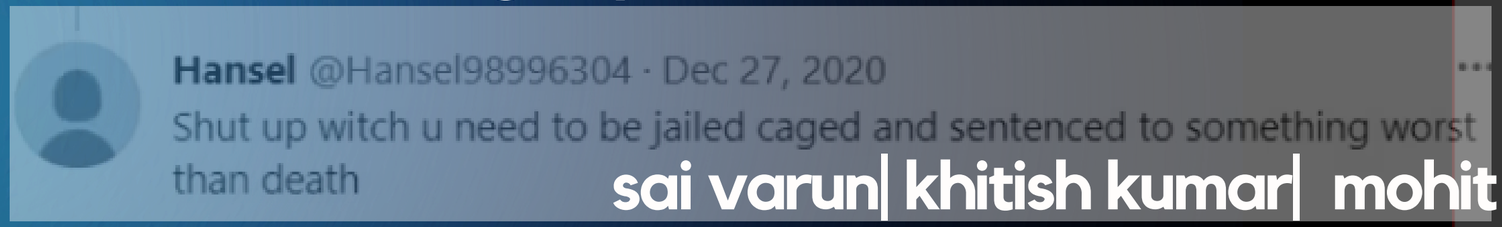


toxic comment and text classification.



group 13

sai varun| khitish kumar| mohit yadav| sai nikhil| charan reddy

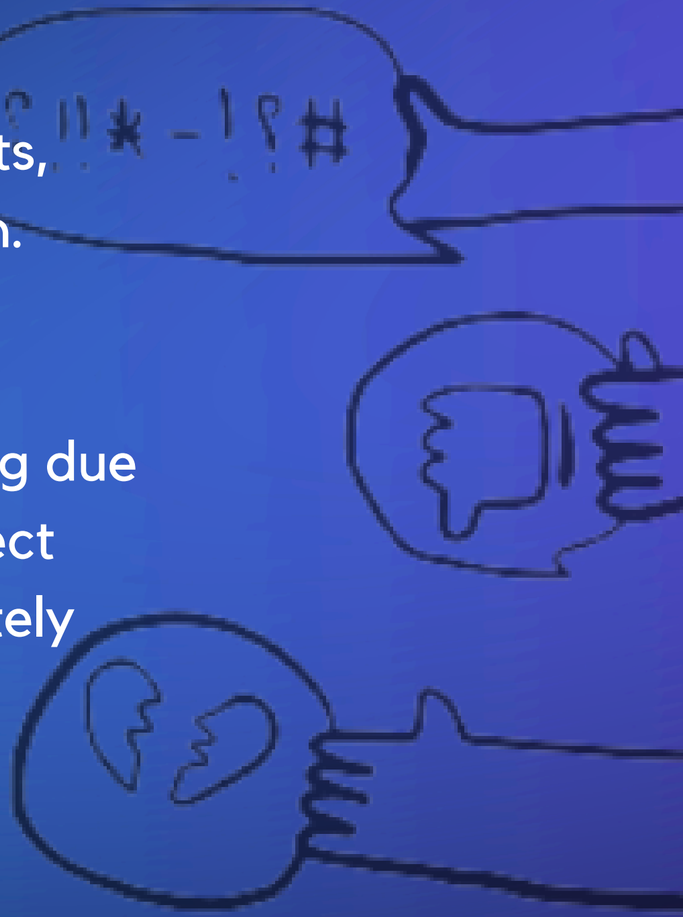
21120| 21121| 21128| 21142| 21149

Table Of Contents

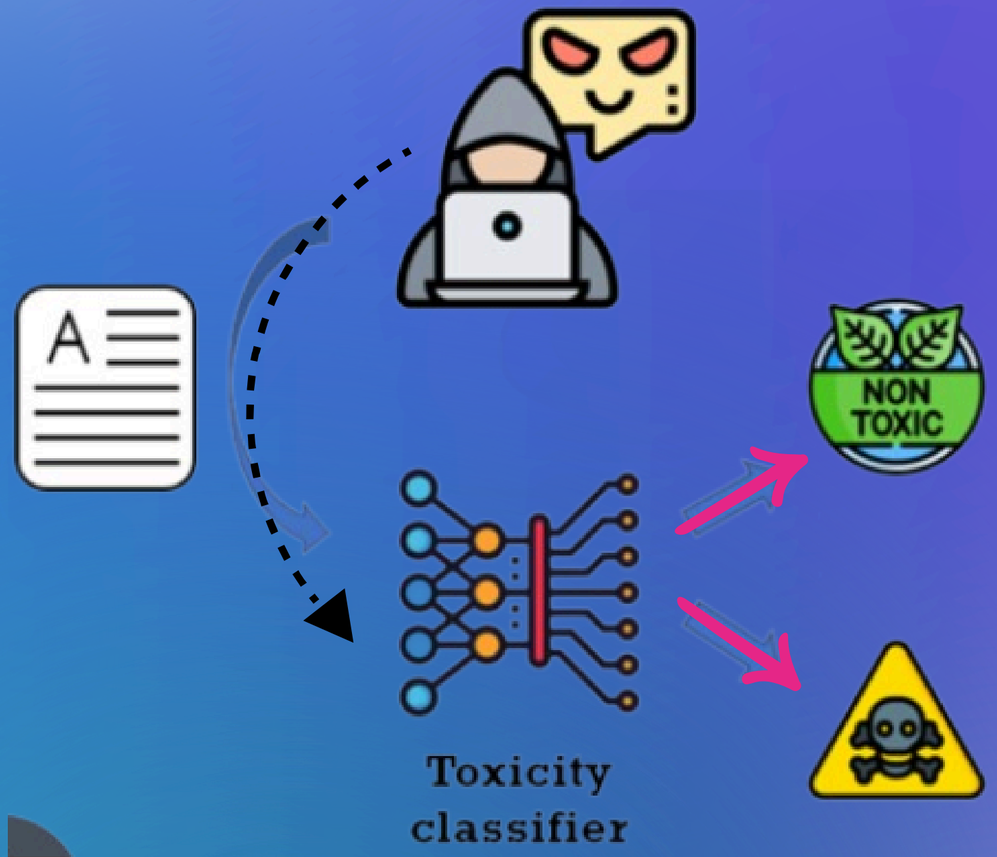

Shut up, nobody cares about your opinion.
This is the dumbest thing I've ever read
I hope you get fired, you're terrible at your job.
toxic comments
Your existence is a waste of space
Go kill yourself.
You're so stupid, I can't believe you think that.
You're such a loser, get a life
You look like a disgusting pig.
Why don't you just disappear
I bet your parents are ashamed of you.
Dumb
Arrogant
Insulting
Vile Nasty
Harass
Ugly
Cruel
Kill
Abusive
Rude
Worthless
Moron
Offensive
Hate
Loser
Disgusting

introduction

Online platforms are often plagued with toxic comments, which can harm user experience and community health. Detecting and mitigating toxic behavior is crucial for maintaining a positive and safe online environment. Managing these toxic comments manually is challenging due to the volume and complexity of the content. This project aims to develop a machine learning model that accurately classifies toxic comments into various categories.



Develop a model capable of accurately classifying various types of toxic comments, including but not limited to toxic, severe_toxic, obscene, threat, insult, and identity_hate. Compare the performance of different language models, including BERT, GloVe, and Bidirectional LSTM, in classifying toxic comments. Evaluate their strengths and weaknesses to determine the most effective model for this task.The ultimate goal is to help online platforms manage and moderate user-generated content effectively, fostering a healthier online community.



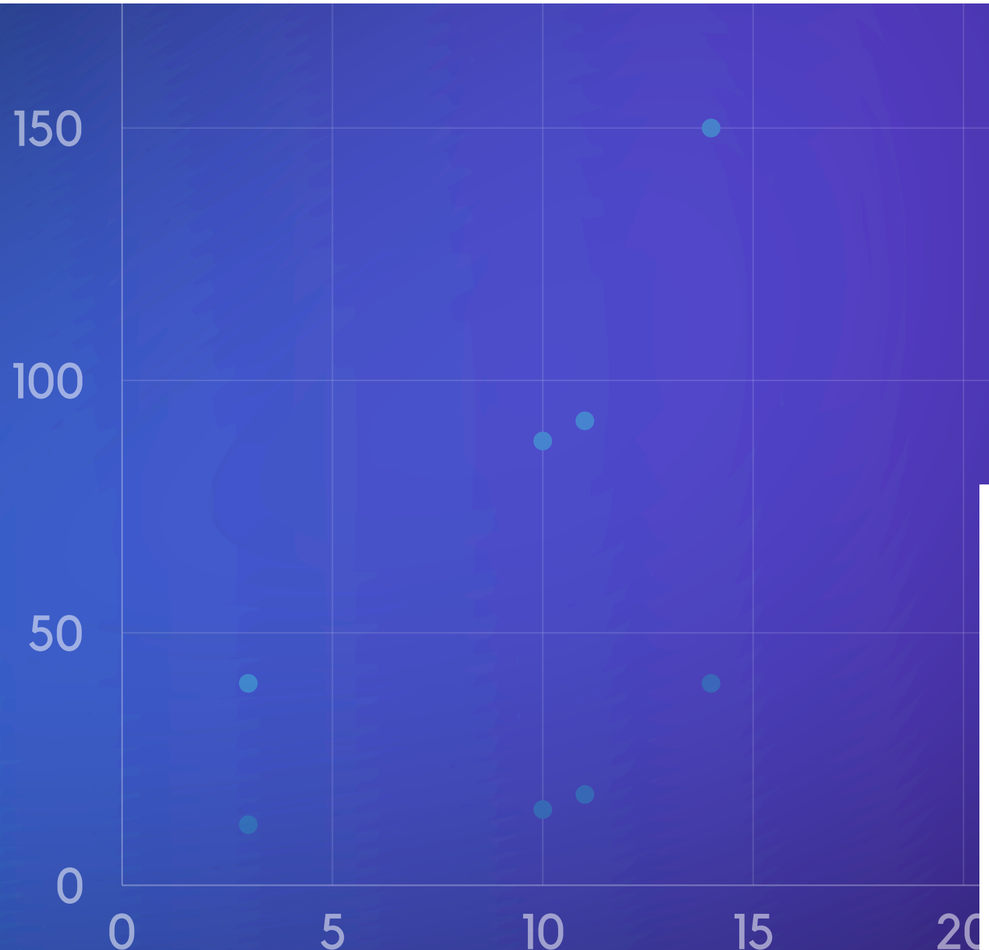
objective.

Series 2 Series 3

```
df_train.head()
```

103...

	id	comment_text	toxic	severe_toxic	obscene	threat	insult	identity_hate
0	0000997932d777bf	Explanation\nWhy the edits made under my usern...	0	0	0	0	0	0
1	000103f0d9cfb60f	D'aww! He matches this background colour I'm s...	0	0	0	0	0	0
2	000113f07ec002fd	Hey man, I'm really not trying to edit war. It...	0	0	0	0	0	0
3	0001b41b1c6bb37e	"\nMore\nI can't make any real suggestions on ...	0	0	0	0	0	0
4	0001d958c54c6e35	You, sir, are my hero. Any chance you remember...	0	0	0	0	0	0



data description

We have dealt with a large number of Wikipedia comments which have been labeled by human raters for toxic behavior.

[106]:

```
df_test.head()
```

[106...

	id	comment_text	toxic	severe_toxic	obscene	threat	insult	identity_hate
0	00001cee341fdb12	Yo bitch Ja Rule is more succesful then you'll...	-1	-1	-1	-1	-1	-1
1	0000247867823ef7	== From RfC == \n\n The title is fine as it is...	-1	-1	-1	-1	-1	-1
2	00013b17ad220c46	" \n\n == Sources == \n\n * Zawe Ashton on Lap...	-1	-1	-1	-1	-1	-1
3	00017563c3f7919a	:If you have a look back at the source, the in...	-1	-1	-1	-1	-1	-1
4	00017695ad8997eb	I don't anonymously edit articles at all.	-1	-1	-1	-1	-1	-1

“toxic”

“severe_toxic”

“obscene”

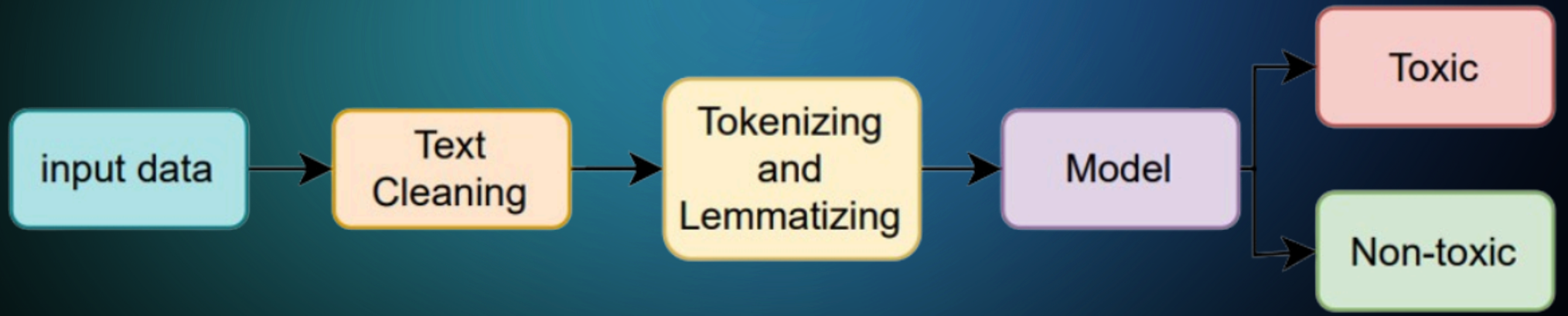
“insult”

“threat”

“identity_hate”

types of toxicity
in data

methodologies.



models used

bert

glove

bidirectional LSTM

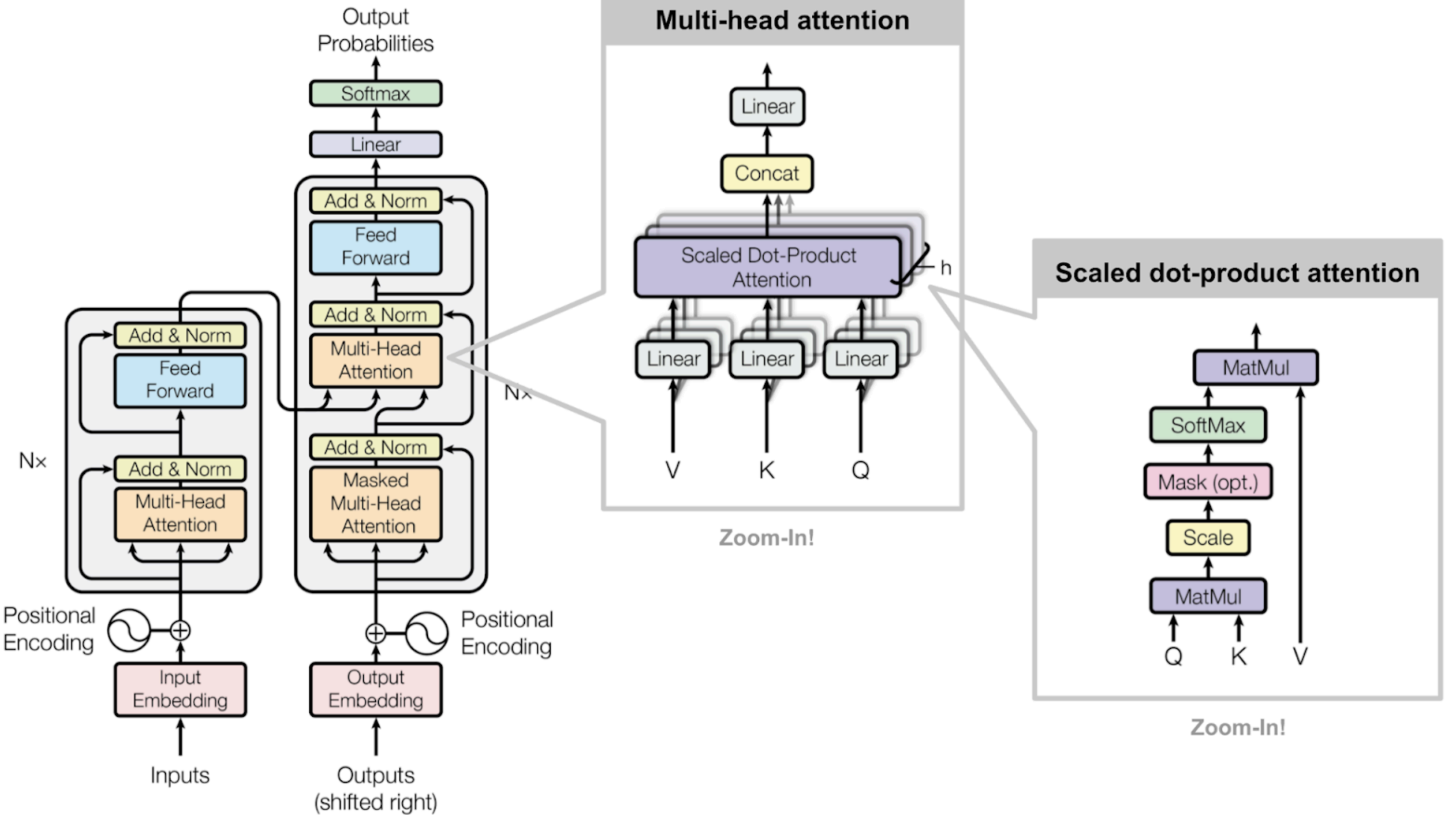
bert

purpose

BERT is used for its advanced natural language understanding capabilities. It provides deep contextual understanding by considering both preceding and succeeding words, crucial for identifying subtle toxic language. BERT excels at capturing nuances in comments, making it highly effective for classifying various types of toxicity. This leads to more accurate and reliable toxic comment detection.

working

BERT is designed to generate a language model so, only the encoder mechanism is used. Sequence of tokens are fed to the Transformer encoder. These tokens are first embedded into vectors and then processed in the neural network. The output is a sequence of vectors, each corresponding to an input token, providing contextualized representations. As opposed to directional models, which read the text input sequentially (left-to-right or right-to-left), the Transformer encoder reads the entire sequence of words at once. Therefore it is considered bidirectional, though it would be more accurate to say that it's non-directional. This characteristic allows the model to learn the context of a word based on all of its surroundings (left and right of the word).


Output Probabilities
Softmax
Linear
Add & Norm
Feed Forward
Add & Norm
Multi-Head Attention
Add & Norm
Masked Multi-Head Attention
Add & Norm
Feed Forward
Add & Norm
Multi-Head Attention
Nx
Nx
Positional Encoding
Positional Encoding
Input Embedding
Output Embedding
Inputs
Outputs (shifted right)
Multi-head attention
Linear
Concat
Scaled Dot-Product Attention
h
Linear
Linear
Linear
V
K
Q
Zoom-In!
Scaled dot-product attention
MatMul
SoftMax
Mask (opt.)
Scale
MatMul
Q
K
V
Zoom-In!

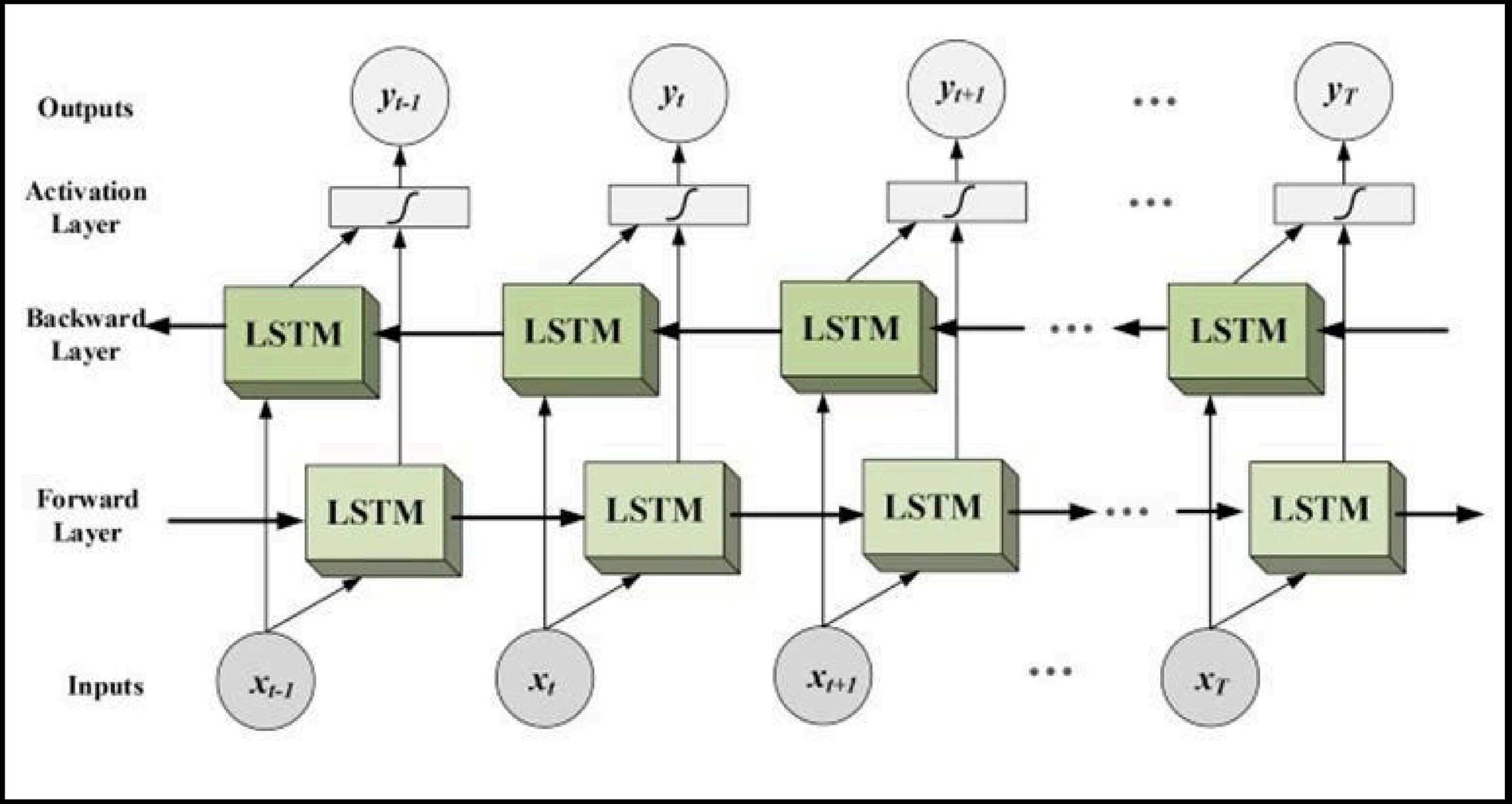
working

Bidirectional LSTM networks process the text data in two directions simultaneously: forward and backward through time steps. They use word embeddings to represent each word and maintain memory states to capture context and dependencies. By combining outputs from both directions, they provide a comprehensive understanding of the comment's context. This approach enhances the model's ability to classify toxic comments accurately by leveraging sequential information and long-term dependencies in the data.

bidirectional Lstm

purpose

Bidirectional LSTMs capture the sequential and contextual information of text by processing it from both directions. This allows for a deeper understanding of word context and long-term dependencies, crucial for identifying nuanced toxic language. Their ability to detect patterns and dependencies in text improves the accuracy of toxic comment classification.


Outputs
Activation Layer
Backward Layer
Forward Layer
Inputs
y_{t-1}
y_t
y_{t+1}
y_T
LSTM
x_{t-1}
x_t
x_{t+1}
x_T

glove

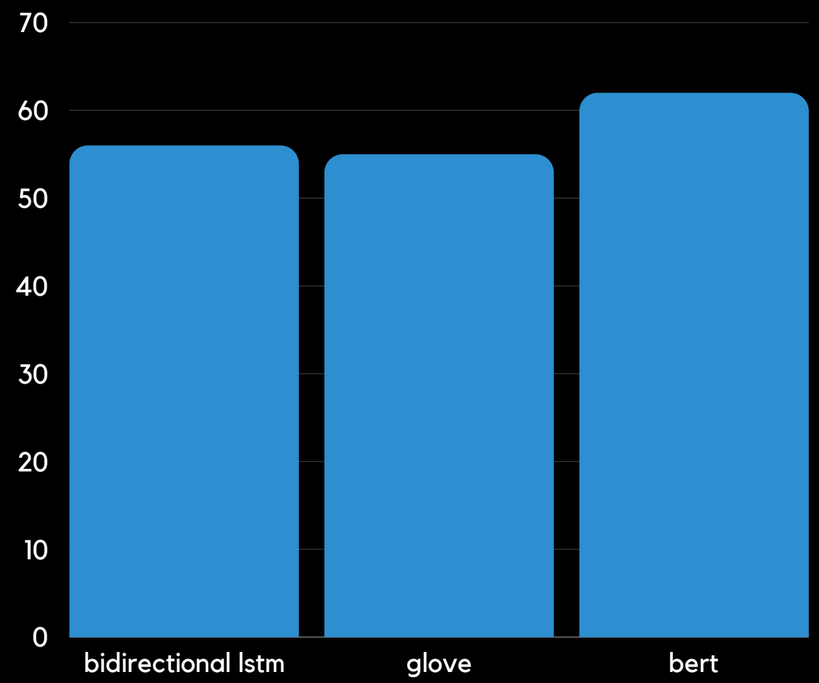
purpose

GloVe creates dense vector representations of words based on their co-occurrence in a large corpus, capturing semantic relationships. These pre-trained word embeddings help models understand the meaning and context of words, improving classification accuracy. GloVe efficiently handles vocabulary variations and slang, and its pre-learned linguistic knowledge enhances model performance in detecting toxic comments.

working

GloVe (Global Vectors for Word Representation) converts words into numerical vectors that capture their meanings and relationships by analyzing word co-occurrences in a large text corpus. This technique helps in understanding word context by creating vectors where similar words are placed close to each other. In toxic comment classification, GloVe vectors are used to represent the words in comments, allowing neural networks, such as Bidirectional LSTMs, to process these vectors and understand the context of the comments. This enables the model to accurately classify comments as toxic or non-toxic and identify the type of toxicity. Using GloVe embeddings improves the model's ability to grasp word relationships, enhancing classification accuracy.

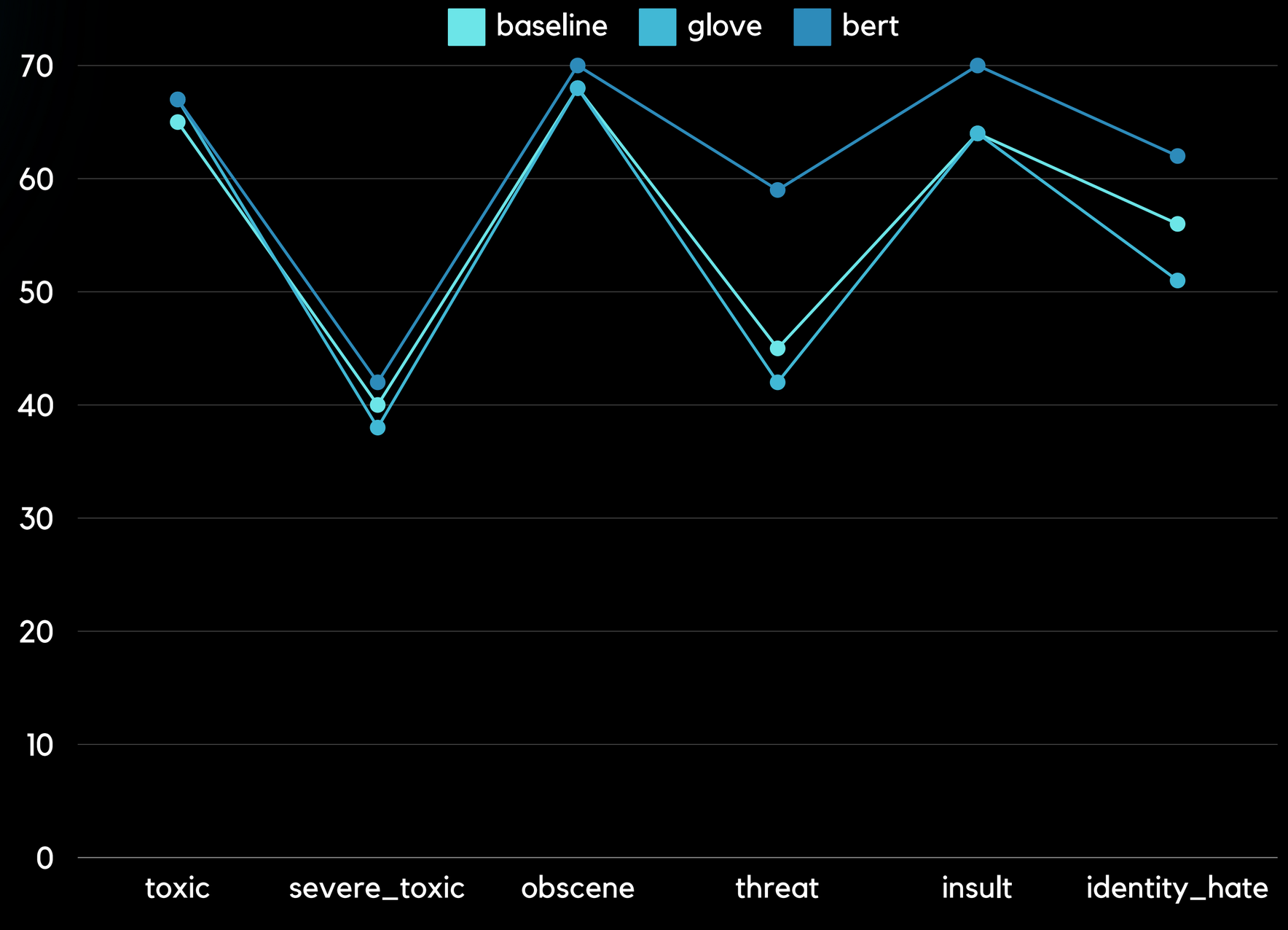
results



F1 macro avg

The BERT model is the best choice for your toxic comment classification task, as it provides the highest F1 macro average score, indicating better overall performance in identifying both toxic and non-toxic comments.

results


baseline
glove
bert
70
60
50
40
30
20
10
0
toxic
severe_toxic
obscene
threat
insult
identity_hate

```
text = 'Heyy blacky go kill yourself'
result = predict_user_input(input_text=text, model=model, tokenizer=tokenizer, device=device)
print(result)
```

✓ 0.5s

```
{'toxic': 1, 'severe_toxic': 1, 'obscene': 1, 'threat': 1, 'insult': 1, 'identity_hate': 1}
```

```
text = 'you look like a disgusting pig'
result = predict_user_input(input_text=text, model=model, tokenizer=tokenizer, device=device)
print(result)
```

✓ 0.2s

```
{'toxic': 1, 'severe_toxic': 0, 'obscene': 0, 'threat': 0, 'insult': 1, 'identity_hate': 0}
```

```
text = 'Go kill yourself'
result = predict_user_input(input_text=text, model=model, tokenizer=tokenizer, device=device)
print(result)
```

✓ 0.2s

```
{'toxic': 1, 'severe_toxic': 0, 'obscene': 0, 'threat': 1, 'insult': 0, 'identity_hate': 0}
```

```
text = 'you look like a disgusting pig'
result = predict_user_input(input_text=text, model=model, tokenizer=tokenizer, device=device)
print(result)
```

✓ 0.2s

```
{'toxic': 1, 'severe_toxic': 0, 'obscene': 0, 'threat': 0, 'insult': 1, 'identity_hate': 0}
```

result gallery.

conclusion.

In conclusion, this project demonstrates the effectiveness of advanced machine learning techniques, including BERT, GloVe embeddings, and Bidirectional LSTM networks, in accurately classifying toxic comments on social media platforms. By leveraging these models, we have achieved significant improvements in identifying various types of toxicity such as toxic, severe_toxic, obscene, threat, insult, and identity_hate. This capability is crucial for enhancing content moderation efforts, promoting healthier online interactions, and ultimately fostering a safer and more inclusive digital community. Moving forward, continued research and development in this area will further refine these models, making them even more robust and adaptable to evolving online behaviors and language nuances.

nothing ruins a day more than getting a toxic comment