# SENTIMENT ANALYSIS

A Project Report
in partial fulfillment of the degree

**Bachelor of Technology**
**in**
**Computer Science & Artificial Intelligence**
**By**

| Roll No | Name of the Student |
|---|---|
| 2203A52057 | T. NAVYA LOHITHA |

**School of Computer Science & Artificial Intelligence**
SR University, Ananthsagar, Hasanparthy (M), Warangal,
Telangana 506371, India

2024-25

# DATASET DESCRIPTION

## DATASET DESCRIPTION

The data for this project is a collection of Amazon product reviews, with every review being tagged as either positive or negative. The data was initially in compressed format and came with two different files: a training set and a test set. Every entry has a customer review and its corresponding sentiment tag. In preprocessing, the labels were changed to numeric values, and the text was processed for analysis. A portion of this vast dataset was utilized in training and testing the models. Its variety and quantity make it extremely effective in creating and testing sentiment analysis systems.

## MOTIVATION AND BACKGROUND

With the rapid growth of online platforms, user-generated content such as reviews, comments, and feedback has become a valuable source of information. Understanding the sentiment behind this content helps businesses improve their products, services, and customer experience. Traditional machine learning approaches require manual feature engineering and often struggle with understanding the context in language. Deep learning models, especially LSTM, CNN, and GRU, have shown significant promise in handling sequential data and capturing complex language patterns. This project is motivated by the need to build a robust sentiment analysis system using these advanced models to achieve high accuracy and generalization in real-world scenarios.

## CONTENT

- **Dataset Preparation:**
  The Amazon Reviews dataset is preprocessed by cleaning the text, encoding sentiment labels, and splitting the data into training and testing sets.
- **Model Development:**
  Three different deep learning models—CNN (Convolutional Neural Network), LSTM (Long Short-Term Memory), and GRU (Gated Recurrent Unit)—are designed and implemented using TensorFlow/Keras. Each model includes an embedding layer for representing words as vectors.
- **Training and Evaluation:**
  The models are trained using the processed dataset. Their performance is evaluated based on accuracy and other metrics like precision, recall, and F1-score.
- **Result Analysis:**
  A comparative study of all models is conducted to determine the most effective architecture. LSTM achieved the highest accuracy, demonstrating superior capability in understanding text sequences.
- **Conclusion:**
  Based on the evaluation, LSTM is identified as the most suitable model for sentiment analysis. The project concludes with insights on model performance and suggestions for future work.

## KEY FEATURES

- **Large-Scale Dataset:**
  The dataset contains millions of real-world product reviews, making it ideal for training deep learning models.
- **Binary Sentiment Classification:**
  Each review is labeled as either positive or negative, enabling straightforward classification.
- **Comprehensive Model Comparison:**
  Multiple deep learning architectures (CNN, LSTM, GRU) are implemented and evaluated for performance.
- **Effective Text Preprocessing:**
  Reviews are cleaned and standardized to ensure meaningful input for the models.
- **Scalable and Generalizable:**
  The models are designed to handle large-scale data and can be extended to other sentiment-based tasks or datasets.

## METHODOLOGY FOR SENTIMENT ANALYSIS

**1. Data Preprocessing:**

The original review texts are preprocessed by eliminating extra characters, making all text lowercase, and eliminating stop words. Sentiment labels are also translated into numerical form for training purposes.

**2. Tokenization and Padding:**

Text data is tokenized, representing words as integer sequences. Sequences are padded to a fixed size to have equal input size.

**3. Word Embeddings:**

An Embedding layer is employed in every model to map tokens to dense vector representations that embody semantic meaning.

**4. Model Architectures:**

**CNN:** Performs local feature extraction from text sequences with 1D convolution and pooling layers.

**LSTM:** Records long-term dependencies and context in the text with memory cells.

**GRU:** Equivalent to LSTM but less parameter intensive, hence being quicker and efficient and yet managing sequence patterns.

**5. Training of models:**

All the models are trained employing the binary cross-entropy loss function along with the Adam optimizer. Training and validation splits of the dataset are used in order to observe the performance throughout the training.

**6. Evaluation:**

Models are assessed in terms of accuracy, precision, recall, and F1-score. Test set performance determines which model best generalizes to new data.

# IMPLEMENTATION

The implementation of the sentiment analysis system using deep learning models(CNN, LSTM, GRU) is available in the following GitHub repository:

**Sentiment Analysis GitHub Repository**

This repository contains the Jupyter Notebook titled **"sentiment-analysis-cnn-lstm-gru.ipynb,"** which demonstrates the complete implementation, including:

- **Dataset Preprocessing:**
  Cleaning and processing Amazon product review text data for training sentiment classification models.
- **Model Architecture:**
  Building CNN, LSTM, and GRU models using TensorFlow/Keras with embedding layers for word representation and suitable layers for learning sequence patterns.
- **Training and Validation:**
  Training each model with the prepared dataset, applying dropout for regularization, and monitoring training/validation performance over multiple epochs.
- **Evaluation and Inference:**
  Evaluating model performance using accuracy, precision, recall, and F1-score, and testing the models on unseen data to assess generalization.

# RESULTS

## 1. Naive Bayes Model Performance

- The Logistic Regression model outperforms Naive Bayes with a higher accuracy of 90.08%.
- Both models show balanced performance across both sentiment classes (positive and negative).
- Logistic Regression shows better generalization and precision-recall consistency, making it a strong baseline model for sentiment classification.

```
Naive Bayes Model Performance
Accuracy: 0.8481
              precision    recall  f1-score   support

           0       0.85      0.84      0.85    399907
           1       0.84      0.85      0.85    400093

    accuracy                           0.85    800000
   macro avg       0.85      0.85      0.85    800000
weighted avg       0.85      0.85      0.85    800000

Logistic Regression Model Performance
Accuracy: 0.900755
              precision    recall  f1-score   support

           0       0.90      0.90      0.90    399907
           1       0.90      0.90      0.90    400093

    accuracy                           0.90    800000
   macro avg       0.90      0.90      0.90    800000
weighted avg       0.90      0.90      0.90    800000
```

## 2. LSTM Model Performance

- The model consistently improved across epochs, with increasing accuracy and decreasing loss.
- The minimal difference between training and validation accuracy indicates that the model generalized well without overfitting.
- With a test accuracy of **94.78%**, LSTM outperformed traditional models like Naive Bayes and Logistic Regression.

```
Epoch 1/3
90000/90000 ━━━━━━━━━━━━━━━━━━━━ 830s 9ms/step - accuracy: 0.9213 - loss: 0.1993 - val_accuracy: 0.9431 - val_loss: 0.1501
Epoch 2/3
90000/90000 ━━━━━━━━━━━━━━━━━━━━ 828s 9ms/step - accuracy: 0.9480 - loss: 0.1392 - val_accuracy: 0.9475 - val_loss: 0.1408
Epoch 3/3
90000/90000 ━━━━━━━━━━━━━━━━━━━━ 829s 9ms/step - accuracy: 0.9521 - loss: 0.1293 - val_accuracy: 0.9481 - val_loss: 0.1390
25000/25000 ━━━━━━━━━━━━━━━━━━━━ 103s 4ms/step - accuracy: 0.9480 - loss: 0.1392
LSTM Model Performansı
Accuracy: 0.9478762745857239
```

```
25000/25000 ━━━━━━━━━━━━━━━━━━━━ 95s 4ms/step
Accuracy : 0.9479
F1 Score : 0.9480
Precision: 0.9461
Recall    : 0.9499
```

## 3. CNN Model Performance

- The **Convolutional Neural Network (CNN)** model was trained for **5 epochs** on the sentiment analysis dataset.
- **Training accuracy** improved from **92.35%** (Epoch 1) to **95.82%** (Epoch 5).
- **Validation accuracy** stayed consistently high, peaking at **94.27%**, indicating good generalization.
- Final model evaluation metrics:

  **Test Accuracy**: **94.20%**

  **F1-Score**: **94.21%**

  **Precision**: **95.15%**

  **Recall**: **93.15%**

- he high precision score indicates that the model made **fewer false positive predictions**.
- Overall, the CNN model showed **strong performance and reliability** in classifying sentiment accurately.

```
Epoch 1/5
90000/90000 ━━━━━━━━━━━━━━━━━━━━ 244s 3ms/step - accuracy: 0.9235 - loss: 0.1941 - val_accuracy: 0.9420 - val_loss: 0.1545
Epoch 2/5
90000/90000 ━━━━━━━━━━━━━━━━━━━━ 239s 3ms/step - accuracy: 0.9437 - loss: 0.1502 - val_accuracy: 0.9425 - val_loss: 0.1526
Epoch 3/5
90000/90000 ━━━━━━━━━━━━━━━━━━━━ 239s 3ms/step - accuracy: 0.9499 - loss: 0.1360 - val_accuracy: 0.9410 - val_loss: 0.1577
Epoch 4/5
90000/90000 ━━━━━━━━━━━━━━━━━━━━ 239s 3ms/step - accuracy: 0.9542 - loss: 0.1258 - val_accuracy: 0.9427 - val_loss: 0.1554
Epoch 5/5
90000/90000 ━━━━━━━━━━━━━━━━━━━━ 238s 3ms/step - accuracy: 0.9582 - loss: 0.1167 - val_accuracy: 0.9408 - val_loss: 0.1637
25000/25000 ━━━━━━━━━━━━━━━━━━━━ 40s 2ms/step - accuracy: 0.9421 - loss: 0.1529
📊 CNN Model Performans1
Accuracy: 0.9420074820518494
```

```
25000/25000 ━━━━━━━━━━━━━━━━━━━━ 29s 1ms/step
Accuracy : 0.9420
F1 Score : 0.9414
Precision: 0.9515
Recall   : 0.9315
```

### 4. GRU model performance

- The **Gated Recurrent Unit (GRU)** model was trained over 3 epochs on the sentiment analysis dataset.
- Training accuracy progressed from **92.43%** (Epoch 1) to **95.05%** (Epoch 3), showing steady improvement.
- Validation accuracy remained strong across epochs, with a peak of **94.66%**, indicating effective generalization.
- Final evaluation metrics on the test set were:

  **Test Accuracy: 94.58%**

  **F1-Score: 94.58%**

  **Precision: 94.68%**

  **Recall: 94.48%**

- The balanced F1-score and recall indicate the model is both precise and sensitive in identifying sentiment correctly.
- Overall, the GRU model delivered robust and consistent results, making it a strong candidate for sentiment classification.

```
Epoch 1/3
90000/90000 ━━━━━━━━━━━━━━━━━━━━ 797s 9ms/step - accuracy: 0.9243 - loss: 0.1893 - val_accuracy: 0.9457 - val_loss: 0.1440
Epoch 2/3
90000/90000 ━━━━━━━━━━━━━━━━━━━━ 800s 9ms/step - accuracy: 0.9481 - loss: 0.1383 - val_accuracy: 0.9466 - val_loss: 0.1415
Epoch 3/3
90000/90000 ━━━━━━━━━━━━━━━━━━━━ 796s 9ms/step - accuracy: 0.9505 - loss: 0.1327 - val_accuracy: 0.9460 - val_loss: 0.1431
25000/25000 ━━━━━━━━━━━━━━━━━━━━ 104s 4ms/step - accuracy: 0.9462 - loss: 0.1420
📊 GRU Model Performans1
Accuracy: 0.9458462595939636
```

```
25000/25000 ━━━━━━━━━━━━━━━━━━━━ 95s 4ms/step
Accuracy : 0.9458
F1 Score : 0.9458
Precision: 0.9468
Recall   : 0.9448
```

# CONCLUSION

## Key Points:

1. **GRU Model Performance**
    - Achieved the highest test accuracy of **94.58%**.
    - Balanced F1 Score (**94.58%**), Precision (**94.68%**), and Recall (**94.48%**).
2. **CNN Model Performance**
    - Test accuracy reached **94.20%**.
    - Precision was particularly strong at **95.15%**, with an F1 Score of **94.21%** and Recall of **93.15%**.
3. **Training Stability**
    - Both models showed steadily increasing training accuracy and consistent validation accuracy, indicating **no overfitting**.
4. **Effective Generalization**
    - High performance on test data demonstrates strong **generalization capabilities** to unseen samples.
5. **Balanced Metrics**
    - F1 scores and recall across both models were well-balanced, indicating a **low rate of false positives and false negatives**.
6. **Optimization Potential**
    - Minor improvements (e.g., hyperparameter tuning or ensemble methods) could push model performance even further.

**Conclusion Paragraph**:
The sentiment analysis models constructed in this project performed well for various types, such as CNN and GRU. The GRU model performed the best with a test accuracy of 94.58%, and the CNN model came in second at 94.20%. Both models performed well at identifying the correct sentiment. Their precision and recall values were high and even, indicating the models made hardly any errors. While being trained, the models learned gradually and did not exhibit signs of overfitting. This indicates that they are able to comprehend and perform with new, unseen data. On the whole, these models are trustworthy for actual sentiment analysis. With minor tweaks such as modifying the settings or blending models, the performance might become even better.