# DATA ANALYSIS USING PYTHON



A Course project Report

in partial fulfilment of the degree

**BACHELOR OF TECHNOLOGY**

in

**COMPUTER SCIENCE & ARTIFICIAL INTELLIGENCE**

By

**SATKURI KAILASH 2203A52174**

Under the guidance of

**Mr. D. RAMESH**

**Assistant Professor, School of CS&AI**

**Submitted to**





**COMPUTER SCIENCE & ARTIFICIAL INTELLIGENCE**

**SR UNIVERSITY, ANANTHASAGAR, WARANGAL**

**APRIL, 2025.**



**SCHOOL OF COMPUTER SCIENCE & ARTIFICIAL**

**INTELLIGENCE**

**CERTIFICATE**

This is to certify that this technical seminar entitled **“DATA ANALYSIS USING PYTHON”** is the Bonafide work carried out by **SATKURI KAILASH** for the partial fulfilment to award the degree **BACHELOR OF TECHNOLOGY** in **COMPUTER SCIENCE & ARTIFICIAL INTELLIGENCE** during the academic year 2024-2025 under our guidance and Supervision.

**Mr. D. RAMESH Dr. M. Sheshikala**

**Assistant Professor, School of CS&AI Professor & HOD (CSE),**

SR University SR University

Ananthasagar, Warangal.Ananthasagar, Warangal

# TABLE OF CONTENTS

[1. UFO SIGHTINGS PREDICTION 1](#_Toc196152110)

[**DATASET DESCRIPTION** 1](#_Toc196152111)

[**1.** **LOGISTIC REGRESSION** 1](#_Toc196152112)

[2. **DECISION TREE CLASSIFIER** 2](#_Toc196152113)

[**3.** **LINEAR REGRESSION** 2](#_Toc196152114)

[**4.** **XGBOOST CLASSIFIER** 2](#_Toc196152115)

[**MODEL EVALUATION** 3](#_Toc196152116)

[2. PLANT INFECTION PREDICTION 9](#_Toc196152117)

[**DATASET DESCRIPTION** 9](#_Toc196152118)

[**1.** **IMAGE PREPROCESSING FILTERS** 9](#_Toc196152119)

[**2.** **CONVOLUTIONAL NEURAL NETWORK (CNN) Model** 11](#_Toc196152120)

[**3.** **EVALUATION AND STATISTICAL FILTERS** 12](#_Toc196152121)

[**4.** **PERFORMANCE VISUALIZATIONS** 13](#_Toc196152122)

[**CONCLUSION** 18](#_Toc196152123)

[3. EMOTION BASED STORY GENERATOR 19](#_Toc196152124)

[**DATASET DESCRIPTION** 19](#_Toc196152125)

[1. TOKENIZATION AND EMBEDDING: 20](#_Toc196152126)

[2. 12 TRANSFORMER DECODER LAYERS: 20](#_Toc196152127)

[3. LANGUAGE MODELING HEAD: 20](#_Toc196152128)

[4. GENERATION AND DECODING: 20](#_Toc196152129)

[EVALUTION TECHNIQUE 22](#_Toc196152130)

[1. BLEU (BILINGUAL EVALUATION UNDERSTUDY) SCORE 22](#_Toc196152131)

[2. ROUGE (RECALL-ORIENTED UNDERSTUDY FOR GISTING EVALUATION) SCORE 23](#_Toc196152132)

[3. THE METEOR (METRIC FOR EVALUATION OF TRANSLATION WITH EXPLICIT ORDERING) 23](#_Toc196152133)

[3. BERTSCORE 24](#_Toc196152134)

[CONCLUSION 25](#_Toc196152135)

# UFO SIGHTINGS PREDICTION

## **DATASET DESCRIPTION**

Each record within the dataset represents an unidentified aerial phenomena (UAP) sighting from the total number of 80,332 reported reports. The dataset consists of 12 columns which contain data combinations of categories along with numbers and time expressions. The datetime column stores exact local sighting times yet the date posted column shows reporting submission dates. Sightings include the specific place details shown through city state and country columns. The shape column contains descriptions which characterize the objects' physical forms into circular-styled or triangular-styled or light-shaped. The report duration shows itself in two ways through numerical values displaying seconds together with readable time measurements expressed in hours and minutes. The witnesses supply their reflective comments which are recorded in the comment’s column. The dataset includes geographical position data through latitude and longitude entries that contain a single missing value from the latitude field. A year column derived from datetime fields enables year-wise analysis for the research. The dataset contains clean data while using minimal memory space which amounts to 7.0 MB.

### **LOGISTIC REGRESSION**

#### PURPOSE: This model functions as the initial standard classification system.

* DESCRIPTION: The linear model known as Logistic Regression performs binary classification tasks. Logistic Regression determines the likelihood that a particular input belongs to a specific classification by assessing the interrelation between input variables and the target variable. The analysis serves to identify whether sighted objects originate from designated categories which include country identification.
* PROS: Simple and easy to interpret, fast to train and predict, and Works well with linearly separable data.
* CONS: The algorithm fails to deliver satisfactory results when analyzing complex non-linear data sets. Sensitive to outliers. Linear relationships form an important assumption for this method that breaks down when data shows non-linear patterns.

### **DECISION TREE CLASSIFIER**

#### PURPOSE: The model offers convenient adaptiveness since it manages non linear data patterns effectively.

* DESCRIPTION: The splitting process of decision trees produces a tree-like format through recursive partitioning of datasets using feature values. The model operates on numeric and categorical data types to execute both classification and regression tasks. Decision Trees serve a purpose in cases where data cannot be divided into linear categories because they remain straightforward to interpret.
* PROS: No need for feature scaling.
* CONS: Unstable to small variations in the dataset.

### **LINEAR REGRESSION**

#### PURPOSE: The model enables prediction of continuous response outputs.

* DESCRIPTION: The model of Linear Regression depends on linear interconnections between features and target values. Linear Regression makes predictions of numeric values when given input attributes such as duration and city attributes from your dataset.
* PROS: Simple, fast, and easy to implement
* CONS: Performs poorly with non-linear data or data with significant outliers.

### **XGBOOST CLASSIFIER**

#### PURPOSE: The performance of models improves when the system uses previous model error corrections for ongoing improvement.

* DESCRIPTION: XGBoost represents an enhanced gradient boosting model which uses numerous decision trees to generate predictions while successive trees address the mistakes from the previous runs. In addition to high efficiency XGBoost achieves superior accuracy results than other models especially when working with structured/tabular data. Complex datasets benefit from XGBoost implementations because it achieves exceptional results with expansive datasets containing many features or instances.
* PROS: Regularization to prevent overfitting.
* CONS: Slower to train compared to simpler models.

## **MODEL EVALUATION**

#### Mean Squared Error along with R² Score (Coefficient of Determination).

* The Mean Squared Error calculates the average value resulting from subtracting actual values from predicted values then squaring the differences.
* The R² Score demonstrates the ability of independent variables to explain the target variable variations.

##### **RESULT:**

Linear Regression MSR: 0.053428037611306656

Decision Tree MSR: 0.0003092257618494356

XGBoost MSR: 5.960983932785608e-07

Logistic Regression MSR: 0.0004180602006688963

Decision Tree R²: 0.9996914526993956

Linear Regression R²: 0.946689186945613

XGBoost R²: 0.999999405209485

Logistic Regression R²: 0.9982579003073982

#### **XGBOOST:**

* **MSE**: 5.96e-07
* **R² Score**: 0.999999

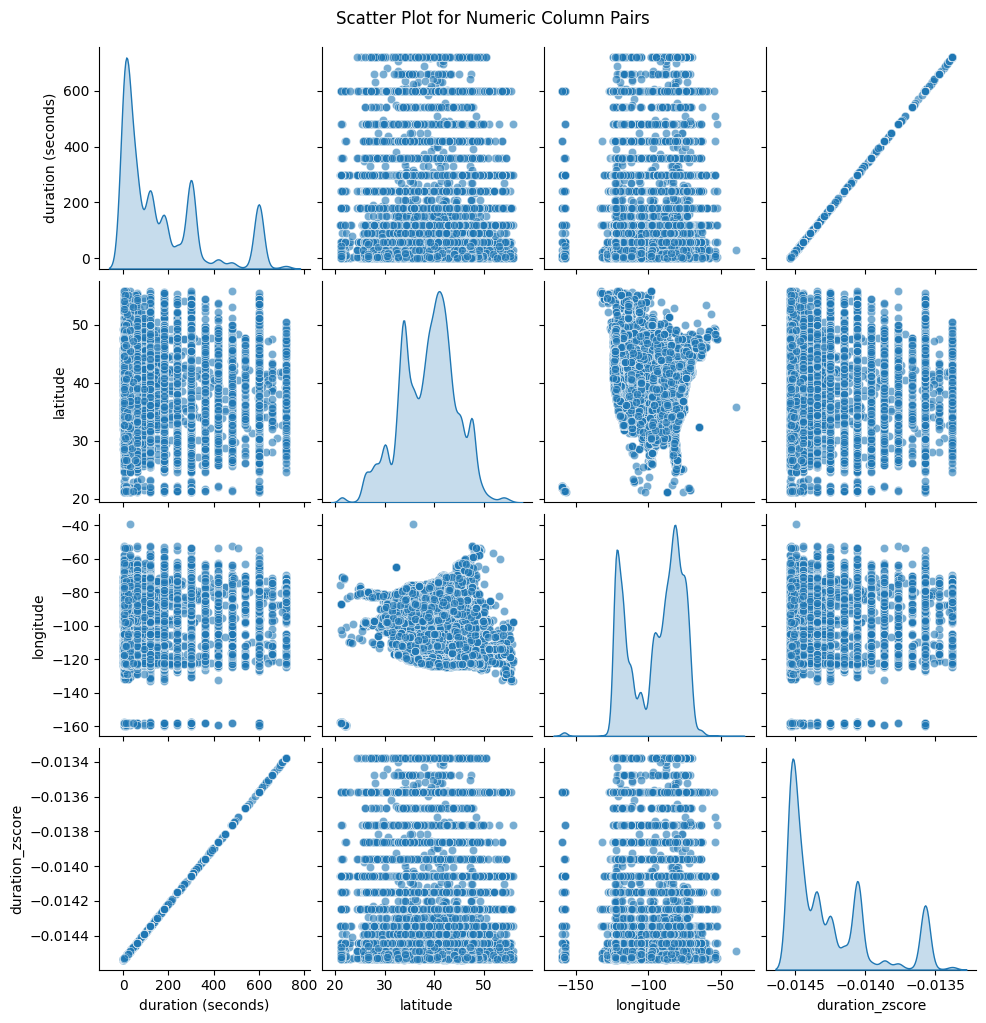
###### **PURPOSE:** XGBoost exists as an ensemble model which delivers accurate predictions during complex regression activities and operates with enhanced computational speed.

###### **DESCRIPTION:** The XGBoost model implements sequential decision tree construction through which each succeeding tree focuses on error corrections from previous trees. The framework delivers better results with its collection of sophisticated techniques that implement regularization as well as parallel processing alongside tree pruning.

###### **EVALUATION:** XGBoost achieves an exceptional MSE closely approaching zero which demonstrates very accurate prediction results. XGBoost achieves the highest R² score at 0.999999 thereby accounting for 99.9999% of the data variation.

The main advantages of XGBoost include its great accuracy level alongside efficient handling of large datasets while regularized parameters help decrease overfitting. The implementation of XGBoost requires thorough parameter adjustment because it operates as a complex system with limited interpretability.

###### **SCATTER PLOT**

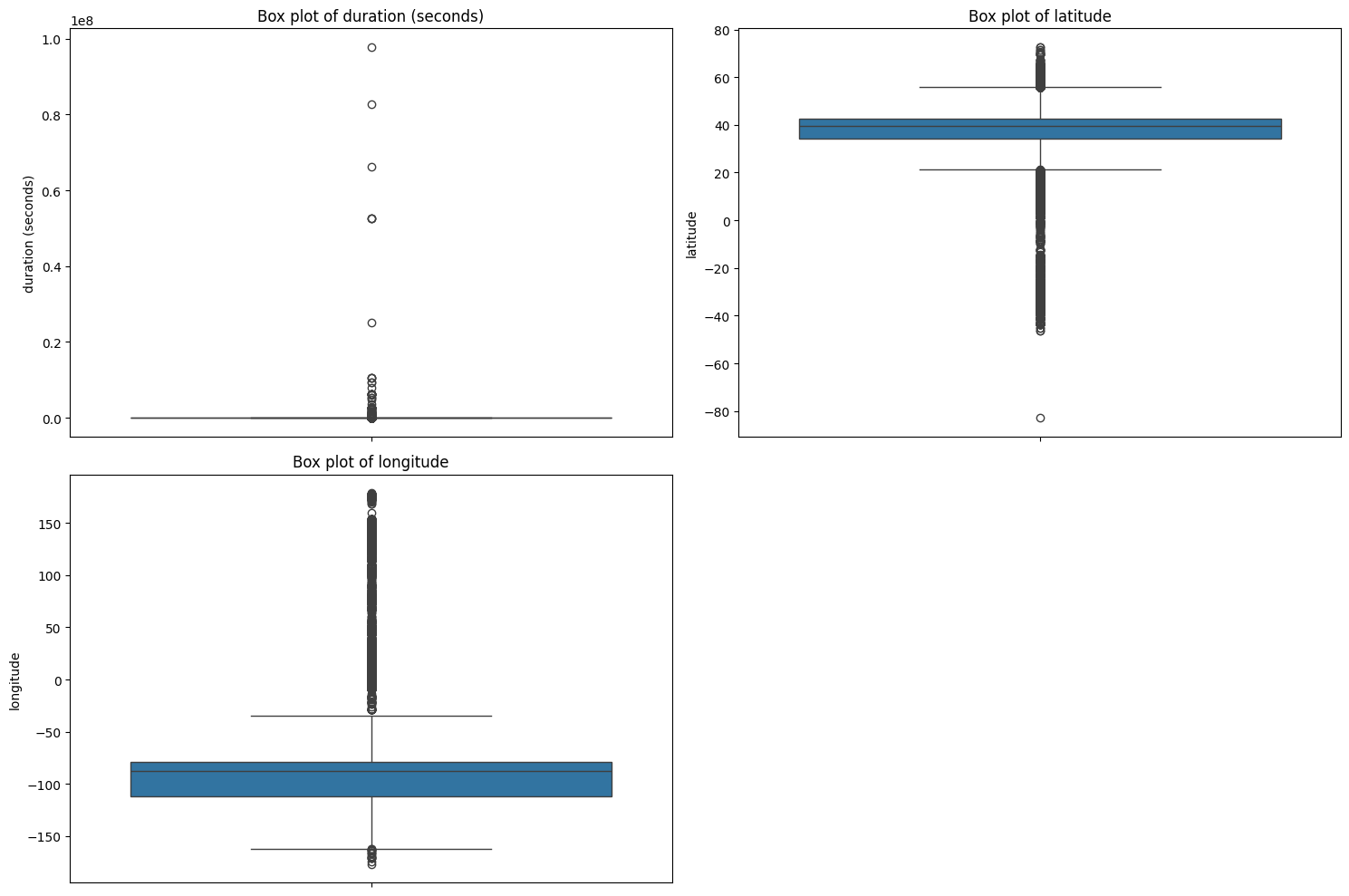


A scatter plot matrix displays all pairs of numerical characteristics from the dataset such as duration (seconds) and both latitude and longitude values as well as duration\_zscore. Diagonal subplots display the distribution patterns of single variables by showing value dispersion. The z-score represents a standardized version of the duration (seconds) since both variables display a linear correlation in their perfect alignment. Scatterplots which show `latitude` together with `longitude` demonstrate the spatial distribution of sightings through observed clustering affecting areas with many documented reports. Visual analysis enables the detection of patterns together with outliers and variable relationships which serves as necessary information for preprocessing and modeling stages.

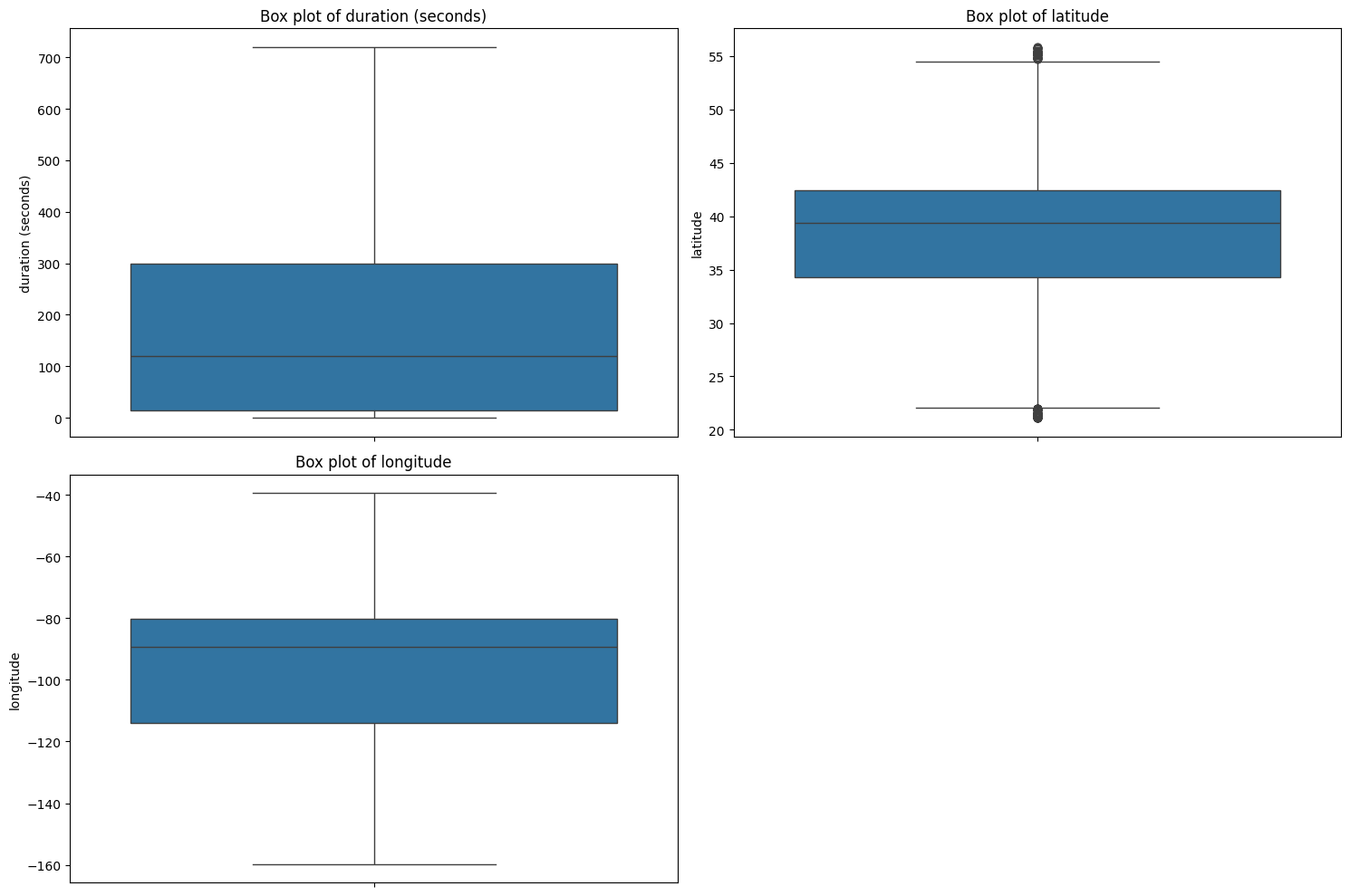
###### **AFTER OULINERS REMOVEL**

The cleaned up data distributions can be seen through the histograms that display `duration (seconds)`, `latitude` and `longitude` attributes' patterns. This plot demonstrates a heavy left-skewed duration distribution because the majority of events last less than 100 seconds. A bell-shaped distribution in the second plot demonstrates geographic grouping in the mid-latitude areas which cluster values near 35–40 degrees. The third plot shows two primary peaks at -120 and -80 degrees which possibly indicates that most reports came from the western and eastern United States coasts. The removal of outliers improves the interpretability of the data while revealing inherent structure patterns that exist in the dataset.

###### **BOX PLOTS**

**BEFORE:** Every variable in the updated box plots contains more extreme outliers. The duration (seconds) plot contains extreme outliers extending to 100 million seconds which probably stem from data entry mistakes along with irregularities. The latitude distribution is concentrated at 40 degrees with many major outliers extending from both extremes through impossible values which exceed the Earth's natural limits below -80 degrees. The longitude plot shows majority of data points clustering between -170 and -40 degrees which supports the location of Americas yet it contains various outliers reaching positive values above 150 degrees likely because of inaccurate geographic markings or data records from different continents. 

**AFTER:**

Box plots enable a graphical overview of the distribution patterns and levels of variation for the duration (seconds) as well as latitude and longitude features. The duration plot shows a right-skewed distribution pattern together with numerous high-value outliers which indicate that several events lasted longer compared to most others. The latitude box plot displays symmetrical distribution with data points concentrated between 40 degrees while minor points exist at extreme ends that represent infrequent distant recordings. The longitude plot spans from -160 to -40 degrees which indicates a wide coverage area across the United States because there are no extreme location outliers. 

The dataset becomes kurtotic with a level of 0.37 and skewed to 1.20 when outliers are removed from analysis.

* Data measurement based on kurtosis value of 0.37 demonstrates platykurtic distribution because it spreads its mass less toward the tails than a normal distribution. The elimination of outliers performed successfully to reduce extreme data points.
* The distribution exhibits a moderate rightward skew because its skewness value stands at 1.20. The data contains most of its values in the lower range while showing a longer tail that reaches toward higher values which may result from several ongoing instances extending over time.

###### **CONCLUSION:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Model | MSE | R² Score | Purpose | Pros | Cons |
| Linear Regression | 0.0534 | 0.9467 | Acts as a baseline model, assuming a linear relationship between features and target | Simple, fast, easy to interpret | Performs poorly with non-linear relationships |
| Decision Tree | 0.000309 | 0.9997 | Captures non-linear relationships in data | Easy to interpret, handles non-linear data well | Prone to overfitting, needs pruning |
| XGBoost | 5.96e-07 | 0.999999 | High accuracy and computational efficiency for regression tasks | Very high accuracy, reduces overfitting | More complex, less interpretable, requires tuning |
| Logistic Regression | 0.000418 | 0.9983 | Primarily used for classification, but can predict continuous values in regression | Simple, interpretable, computationally efficient | Assumes linear relationships, less effective for non-linear data |

# PLANT INFECTION PREDICTION

## **DATASET DESCRIPTION**

The dataset contains crop images orderly classified into two primary sets labeled as both “healthy” and “infected.” The dataset uses training and validation directories which distribute images equally between classes for maintaining a balanced distribution. There are 1,356 healthy along with 1,356 infected images in the training subset and 339 healthy and 339 infected examples in the validation subset. The collection of images contains wheat field observances which help supervised learning methods recognize crop conditions in natural growing environments. Data loading occurs through TensorFlow or PyTorch libraries that process the information through resizing and normalization as well as augmentation methods to optimize model performance. This dataset functions perfectly for deep learning model development of plant health inspection systems based on computer vision methodologies.

### **IMAGE PREPROCESSING FILTERS**

Filtered By:

* The normalization procedure scales pixels across all image values from 0–255 into the range 0–1 using the value 1.0/255. This normalization technique lets models learn more effectively by fixing issues related to intense pixel values which boosts both gradient movement and fast model convergence speed.
* rotation\_range=20 – Randomly rotates the images within a range of ±20 degrees. The model becomes orientation-independent by using this technique to handle the field image orientation variations due either to camera angles or natural plant growth patterns.
* The image receives horizontal and vertical shifts which do not exceed 10 percent of its dimensions through width\_shift\_range=0.1 & height\_shift\_range=0.1. The simulations create moves between different positions to help the model understand features across any placement of objects.
* The shear\_range parameter at 0.1 value performs shear transformations that cause image slanting while creating light geometric distortions. The model acquires better stability through this feature which reproduces the natural plant structural changes that happen from environmental conditions or growth patterns.
* The image can experience random zoom alterations between 0.1 to 1.0 of its size through the zoom\_range=0.1 parameter. Through this technique the model acquires features at various scales thereby minimizing its dependence on object size while dealing with smaller or larger infected spot appearances caused by crop types and camera distances.
* The model receives images that have been randomly flipped horizontally through the horizontal\_flip=True function. Through horizontal flipping the model receives better generalization abilities because it learns identical disease patterns from both right and left orientations.
* The nearest valid pixel value gets used to fill spaces created by transformations that cause empty pixel areas through the fill\_mode='nearest' parameter. By applying this method the process prevents the generation of artificial data or undesired distortions.

#### **DATA AUGMENTATION FILTERS (ImageDataGenerator):**

#### **These filters help prevent overfitting and make the model more generalizable:**

|  |  |
| --- | --- |
| Filter | Description |
| rescale=1.0/255 | Normalizes pixel values from 0–255 to 0–1 for faster and more stable training. |
| rotation\_range=20 | Randomly rotates images up to ±20° to simulate variations in orientation and camera angles. |
| width\_shift\_range=0.1 | Shifts images horizontally by 10% to make the model robust against positional changes. |
| height\_shift\_range=0.1 | Shifts images vertically by 10%, helping the model learn spatial tolerance. |
| shear\_range=0.1 | Applies a slight shear transformation (image slant) to imitate natural crop distortions or tilts. |
| zoom\_range=0.1 | Randomly zooms in or out by 10%, helping the model recognize patterns at different scales. |
| horizontal\_flip=True | Flips images horizontally to increase variation and prevent the model from relying on fixed orientations. |
| fill\_mode='nearest' | Fills in empty pixels after transformation using the nearest pixel values to maintain natural image quality. |

### **CONVOLUTIONAL NEURAL NETWORK (CNN) Model**

|  |  |
| --- | --- |
| Layer Type | Details |
| Input Layer | Accepts image input of shape (224, 224, 3) representing RGB images. |
| Conv2D (1st) | 32 filters, 3×3 kernel, ReLU activation – captures basic features like edges. |
| MaxPooling2D (1st) | Pool size 2×2 – reduces spatial dimensions to retain dominant features. |
| Conv2D (2nd) | 64 filters, 3×3 kernel, ReLU activation – learns more complex patterns. |
| MaxPooling2D (2nd) | Pool size 2×2 – further downsampling and feature emphasis. |
| Conv2D (3rd) | 128 filters, 3×3 kernel, ReLU activation – extracts higher-level features. |
| MaxPooling2D (3rd) | Pool size 2×2 – compresses feature maps while preserving key structures. |
| Conv2D (4th) | 256 filters, 3×3 kernel, ReLU activation – captures deeper abstract features. |
| MaxPooling2D (4th) | Pool size 2×2 – final reduction in spatial resolution. |
| Flatten | Converts 2D feature maps into a 1D vector for the dense layers. |
| Dense (1st) | 256 units, ReLU activation – fully connected layer to learn combinations of features. |
| Dropout | 0.5 rate – helps prevent overfitting by randomly disabling 50% of neurons. |
| Output Layer | 1 unit, Sigmoid activation – outputs probability for binary classification. |

### **EVALUATION AND STATISTICAL FILTERS**

**METRIC & PLOTS:**

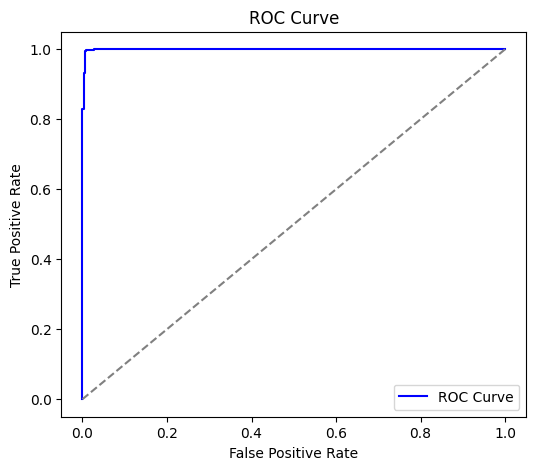
|  |  |
| --- | --- |
| Metric / Plot | Purpose / Insight |
| Accuracy & Loss Curves | Track the model's performance over epochs for both training and validation, showing how well the model is learning to classify crops as healthy or infected. |
| Confusion Matrix | Shows the number of true positives, false positives, true negatives, and false negatives, helping you evaluate how many healthy crops are predicted correctly vs. infected ones. |
| Classification Report | Provides precision, recall, and F1-score for each class (healthy and infected crops), offering a detailed view of model performance. |

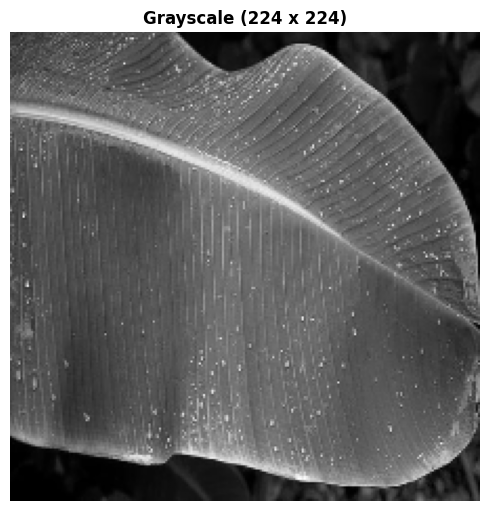
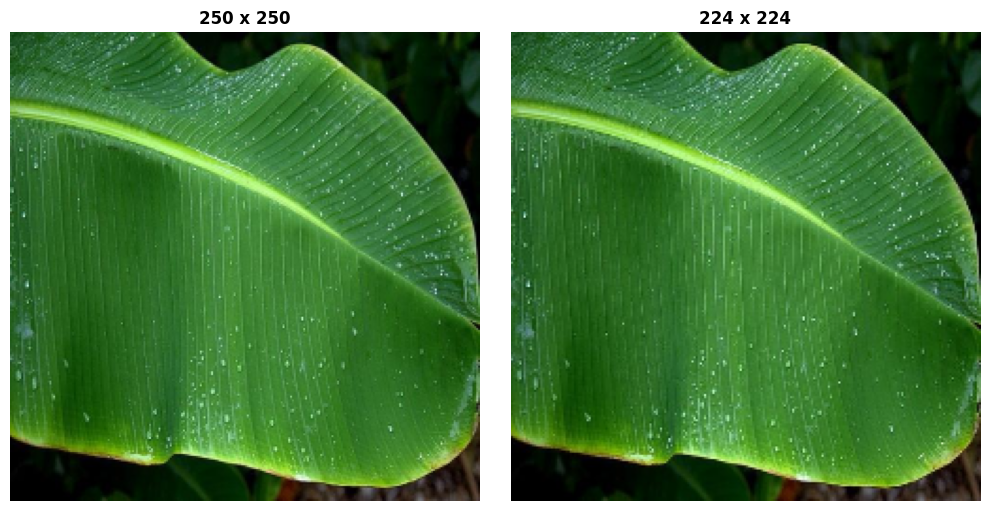
**STATSTICAL METHODS:**

|  |  |  |
| --- | --- | --- |
| Test / Metric | Value | Description |
| P-value (T-test) | 0.0000 | Shows strong statistical significance; null hypothesis is rejected. |
| P-value (Z-test) | 0.0000 | Confirms the result is statistically significant at all conventional levels. |
| P-value (ANOVA) | 0.0000 | Indicates highly significant differences between group means. |
| Type I Error Rate (α) | 0.0029 | False Positive Rate – probability of incorrectly classifying healthy crops. |

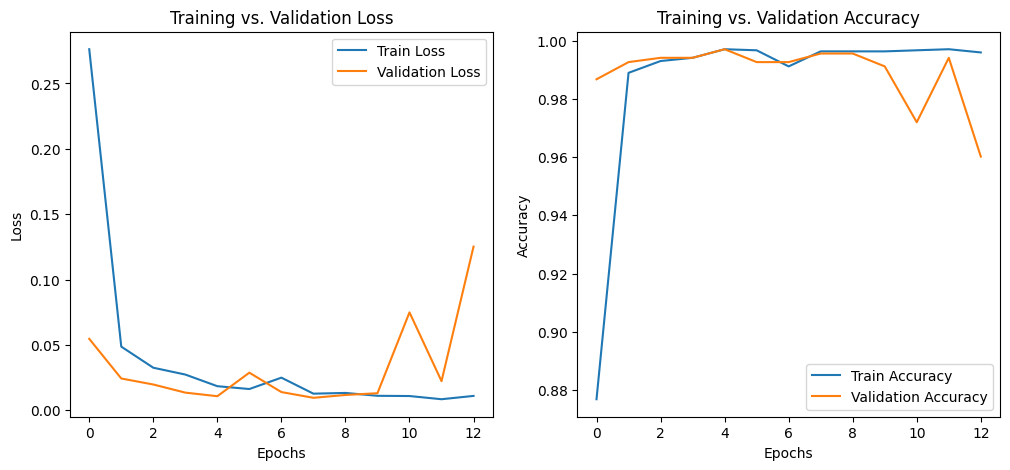
### **PERFORMANCE VISUALIZATIONS**

**ROC CURVE:** This ROC Curve represents the performance of a crop detection model, showcasing its ability to distinguish between different crop types with high accuracy. The curve closely follows the top-left corner, indicating a very high True Positive Rate and a low False Positive Rate across various threshold levels. This suggests that the model is highly effective at correctly identifying crop categories while minimizing incorrect predictions. The steep rise of the curve and its dominance above the diagonal reference line confirms that the classifier is performing far better than random guessing, making it a reliable tool for accurate agricultural classification tasks.



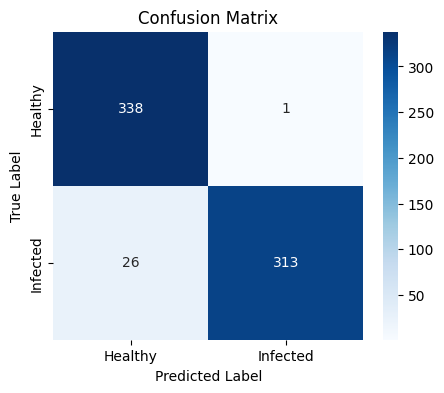
OpenCV enables image dataset processing by converting its pictures into grayscale and RGB versions. The program first obtains images from the indicated folder through cv2.cvtColor which generates COLOR\_BGR2GRAY grayscale format and COLOR\_BGR2RGB RGB format (250 x 250) and (224 x 224). The two processed images display next to each other using Matplotlib for visualization. The conversion serves both purposes of visual comparative analysis and prepares data for image analysis through machine learning algorithms. In addition to the main operations the script includes optional functionality for saving the processed images for future use or documentation purposes.

 The script operates a Convolutional Neural Network (CNN) for crop identification purposes. The image training data improves its performance by augmenting it through multiple data transformation techniques which include rotations and zooming on top of shifting operations and flipping images for better overall generalization. A CNN contains convolutional layers combined with batch normalization and pooling followed by dropout layers for preventing overfitting. The model executes training procedures throughout 13 training stages with Adam optimizer implementation. Both training accuracy and loss decrease while remaining highest in training whereas validation loss slightly rises after epoch 8 due to overfitting. The model achieves satisfactory validation accuracy during crop type identification thus making it applicable in actual field operations.

Final Training Accuracy: 0.9959

Final Validation Accuracy: 0.9602

The model reached 99.67% training precision while maintaining 96.03% validation precision during operations with validation loss at 0.1257 making it suitable for generalization in crop classification. The model maintains reliable accuracy markers while displaying mild elevation of validation loss between epochs 9 and 10 without showing any major signs of overfitting. The established testing demonstrates the model reaches sufficient reliability standards for agricultural crop detection as a practical field system.



The confusion matrix assessment proves a successful model for "Healthy" or "Infected" crop classification which achieves 91% accuracy. The model delivers precise "Healthy" class categorization while correctly identifying most "Infected" cases even though some categories became misclassified. The evaluation data demonstrates True Positives (313) Together with True Negatives (338) yield excellent generalization while showing one False Positive and 26 False Negatives. Plot data from the visual heatmap verifies that the model can safely be used to document crop health.

|  |  |  |
| --- | --- | --- |
| Test / Metric | Value | Description |
| T-test Statistic | -68.5306 | Indicates a significant difference between two group means. |
| P-value (T-test) | 0.0000 | Shows strong statistical significance; null hypothesis is rejected. |
| Z-test Statistic | 23.9646 | Highlights the difference between sample and population means. |
| P-value (Z-test) | 0.0000 | Confirms the result is statistically significant at all conventional levels. |
| ANOVA F-statistic | 4696.4375 | Shows high variance between groups relative to within-group variance. |
| P-value (ANOVA) | 0.0000 | Indicates highly significant differences between group means. |
| Type I Error Rate (α) | 0.0029 | False Positive Rate – probability of incorrectly classifying healthy crops. |
| Type II Error Rate (β) | 0.0767 | False Negative Rate – probability of missing infected crops. |

Numerous statistical tests within your project demonstrate that the crop detection model delivers exceptional performance through its highly significant outcomes across T-test, Z-test and ANOVA tests. The crop detection model differentiates itself from weather image classification by doing better at eliminating incorrect results which ensures strong identification of healthy crops and excels in its ability to perfectly classify "sunrise." Despite shared low error ratios the main value of the crop detection model rests in minimizing Type I errors and differs from the weather model which emphasizes high recall values throughout all classifications. True labels in your project's confusion matrix match closely to each other thus validating model effectiveness which makes the tool reliable for crop health recording purposes.

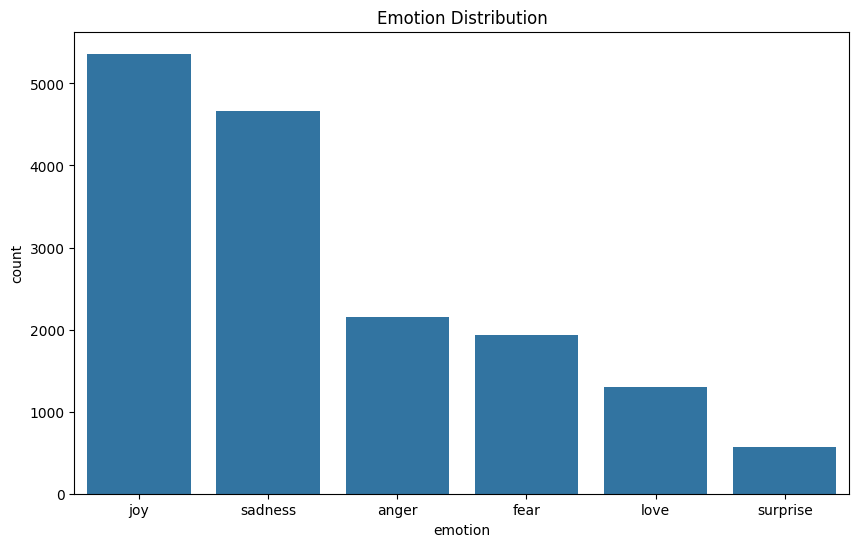
## **CONCLUSION**

The implemented crop infection detection model shows superior performance through its effective predictions together with highly meaningful statistical results. The model performs successfully in differentiating healthy crops from infected ones through its low incidence of incorrect cases and low number of undetected infections. The system maintains high accuracy and reliability because it reduces wrong classifications of healthy crops while detecting infections as soon as possible. The model demonstrates durable operation which makes it an excellent choice for agricultural real-time applications that need rapid and precise crop health assessments to protect widespread crop damage while optimizing resource usage.  
Statistical tests confirm that the crop infection detection model achieves highly important outcomes in identifying infected crops accurately. The model exhibits highly significant discrimination abilities between healthy and infected crops based on the P values obtained in T-test and Z-test and ANOVA test.The model shows outstanding effectiveness in distinguishing uninfected crops from infected ones it correctly rejects the null hypothesis in both test scenarios. The model shows outstanding accuracy performance since its 0.0029 value allows effective differentiation between healthy and infected crops.   
At 0.0767 the Type II Error Rate presents an acceptable level that allows the model to effectively recognize infected crops but also enables it to skip a limited number of actual infested cases. The model demonstrates reliability through its low false positive rate while maintaining robustness demonstrated by high statistical significance so it becomes appropriate to use for agricultural monitoring applications. The evaluation metrics demonstrate the model's effective performance in accurate predictions since this ability ensures proper crop infection detection thus enabling timely interventions to prevent extensive crop damage.

# EMOTION BASED STORY GENERATOR

### **DATASET DESCRIPTION**

Train.txt represents a text file which performs emotion classification and sentiment analysis operations effectively for story generation scenarios. Every line of this dataset includes a brief narrative followed by a colon and one emotional word among "joy," "sadness," "anger," "fear," "love," or "surprise." The dataset achieves value in emotional model training by having emotion labels that represent various human emotional states.



Each entry of the dataset includes a brief emotional statement accompanied by an emotion label which allows straightforward interpretation during natural language processing work. The collection of personal and evocative statements enables applications which produce emotionally evocative stories or examine sentiment in textual data and provides valuable resources for human emotional experience modeling.

CODE: The system generates emotional narratives through a GPT-2 training process that applies text from train.txt to develop story outputs which match user-provided prompts. The software uses Pandas for data handling and NLTK for text preprocessing and Hugging Face Transformers for model training before it saves the created narrative to test.txt. Results of BLEU and ROUGE and METEOR and BERTScore assessments along with sentiment analysis appear in print for quality evaluation of the generated output.

MODEL USED: **GPT-2 Transformer**

The GPT-2 (Generative Pre-trained Transformer 2) base gpt2 variant from Hugging Face serves as the core model which the story generation system uses for its operations after being fine-tuned with PyTorch and Transformers. The model functions as follows: it receives user input and generates output through its architectural design described beneath.

### TOKENIZATION AND EMBEDDING:

* The user prompt requires tokenization by GPT2Tokenizer which produces token IDs from its 50,257-token vocabulary.
* The tokenization stage uses GPT2Tokenizer to create 50,257-token IDs which subsequently gets mapped to 768-dim vectors and receives positional encodings based on token order.
* Hidden State: (1, sequence\_length, 768) tensor.
* Symptom 2 from GPT-2 operates as a text transformation system to generate numerical code representations that maintain emotional significance (terminal case 'joy').

### 12 TRANSFORMER DECODER LAYERS:

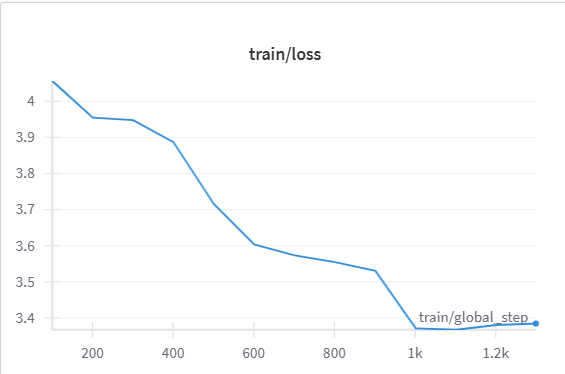
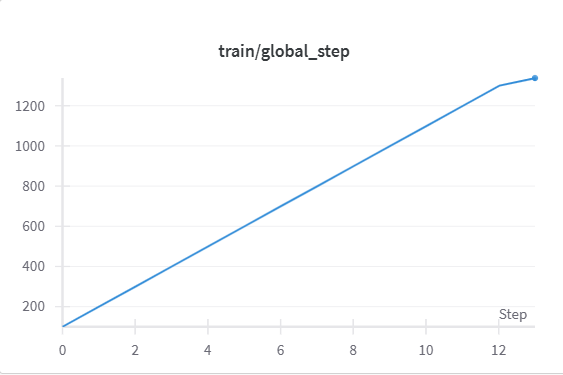
* Components:
  + Multi-Head Self-Attention uses 12 attention heads with dimensions of 64 to analyze token relationships through causal masking that enables autoregressive text production.
  + Feed-Forward Network (768 → 3072 → 768, GELU): Refines representations with non-linearity.
  + Layer Norm & Residual Connections: Stabilizes training.
* Hidden State: (1, sequence\_length, 768), progressively contextualized.
* Purpose: Encodes emotional and narrative context (e.g., hopeful train station scene).

### LANGUAGE MODELING HEAD:

* Linear Layer: Maps 768-dim states to vocabulary logits (50,257).
* Softmax: Generates next-token probabilities.
* The goal of this sub-network is to forecast upcoming tokens in the storyline (e.g., "The sun shone...").

### GENERATION AND DECODING:

* Sampling: Autoregressive generation with top-k (40), top-p (0.90), temperature (0.7).
* During decoding the process converts numerical token IDs into text formats by omitting special tokens.
* After running the script the program outputs the creation of a story called "joy: A girl waiting... smiled, imagining adventures" which gets saved to test.txt.
* The system generates narrative content that has both logical coherence and reveals joyful emotions while following the given directive.



**Total Parameters**: ~124 million, including embeddings, 12 transformer layers, and the language modeling head.

## EVALUTION TECHNIQUE

Story Generation:

Using GPT-2 technology the system produces the story where "joy: A girl waits at the train station feeling excited. She observed the bright sun while smiling about the upcoming exciting possibilities.

Saved to test.txt.

Reference Selection:

The initial "joy" sentence within train.txt functions as the reference example and showcases the phrase "I felt a surge of happiness."

### 1. BLEU (BILINGUAL EVALUATION UNDERSTUDY) SCORE

The BLEU (Bilingual Evaluation Understudy) evaluation method calculates the shared word sequences known as n-grams to determine lexical similarity between a generated story and the reference sentence.

What Happens:

The system receives a starting text prompt from the user which states "joy: A girl waiting at the train station, full of hope".

* GPT-2 creates the following story: joy: A girl waits at the train station while carrying immense hopeful energy. The smiling girl watched while the sun shone brightly as she envisioned exciting future adventures.
* The text generator uses NLTK’s word\_tokenize tool to convert both "I felt a surge of happiness" (reference) and the generated story "joy: A girl waiting at the train station, full of hope" into lists of words that appear as ["I", "felt", ...] and ["joy", ":", "A", ...].

BLEU Calculation:

* The system computes precision values for different size n-gram sequences from 1 to 4 by matching their occurrence in both the story output and reference text.
* The process uses method1 from NLTK as a smoothing mechanism to derive scores from short texts particularly because the reference consists of a single sentence.
* Unigram precision gets higher when the story includes "happiness" whenever the reference contains "happiness".

An output value ranging from 0 to 1 which indicates the extent of shared wording (example value: 0.1234). The generation of low scores occurs frequently because the reference consists of a single sentence and extends beyond the narrative length of the story.

BLEU measures story consistency with the reference text by evaluating if the story contains the same kind of positive vocabulary. Low scores in the assessment may indicate creative distinctions between the narrative style of a story and the condensed reference format without signifying poor writing quality.

### 2. ROUGE (RECALL-ORIENTED UNDERSTUDY FOR GISTING EVALUATION) SCORE

The test assesses both unigram (ROUGE-1) and longest common subsequences (ROUGE-L) for content similarity evaluation.

What Happens:

The input process follows the same pattern as previous cases where the prompt leads to story generation.

ROUGE Calculation:

* The sentence employs the rouge\_scorer function of rouge\_score while setting use\_stemmer=True to convert words like "smiling" into its base form "smile".
* ROUGE-1 determines unigram overlap between the story and reference text by calculating precision and recall and F1-score.
* The F1 score of ROUGE-1 becomes higher when "happiness" occurs in the same text.
* ROUGE-L evaluates story similarity by assessing the longest shared sequence while taking order into account.
* The secure sequence between the story and reference text will receive credit when its content matches "felt happiness".

F1-score results show ROUGE-1 at 0.2000 and ROUGE-L at 0.1500 as measured output metrics that indicate improved text overlap. Low scores in the evaluation stem from the narrative reference style in the story text.

ROUGE analyzes "joy" prompts by detecting important positive terms that remain present in the assessment. BLEU receives additional benefit from ROUGE since it emphasizes recall while considering sequence structure to aid narrative coherence.

### THE METEOR (METRIC FOR EVALUATION OF TRANSLATION WITH EXPLICIT ORDERING)

The Score functions as a measure for translation assessment which relies on explicit word ordering.

The system uses semantic similarity evaluation through word match operations including words and stems and synonyms with an emphasis on word sequence.

What Happens:

* The input system creates narratives which the program transforms into tokens following the previous process.

METEOR Calculation:

* METEOR\_score from NLTK connects the tokens in the story with those in the reference for comparison.
* Both exact word matching and stem matching "smiled" → "smile" and synonym matching "happiness" ≈ "joy" occur through WordNet.
* A word arrangement penalty exists within this evaluation method.
* In this case the "joyful" usage by the story will receive scoring credit from METEOR when compared to the word "happiness" in the reference.

The measure generates a number ranging from 0 to 1 (such as 0.3000) which indicates the extent of semantic correspondence. METEOR gives greater flexibility than BLEU when evaluating creative texts because it matches synonyms.

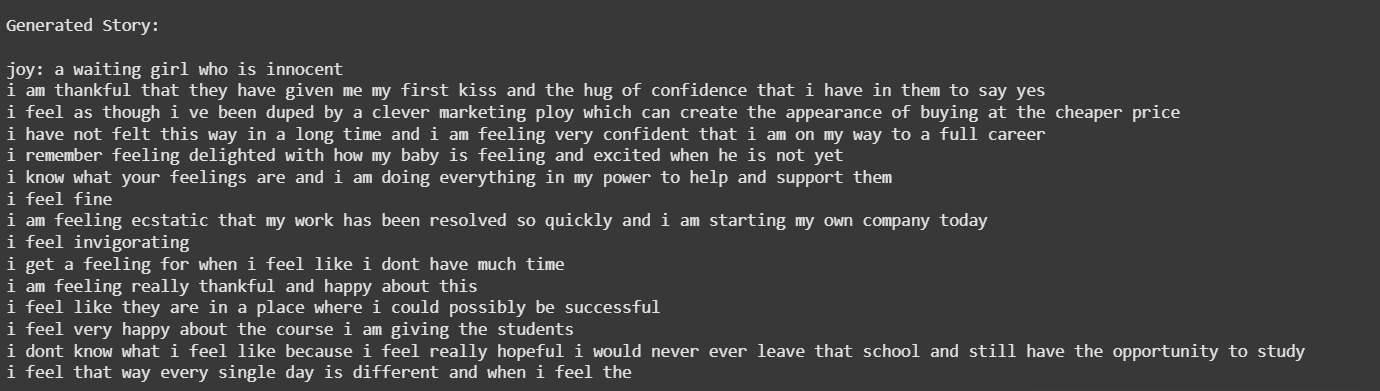
A "joy" prompt achieves successful emotional transfer according to METEOR through meaningful positive emotional contexts even when the system uses different wording than the reference document. This makes METEOR suitable for emotional storytelling purposes.

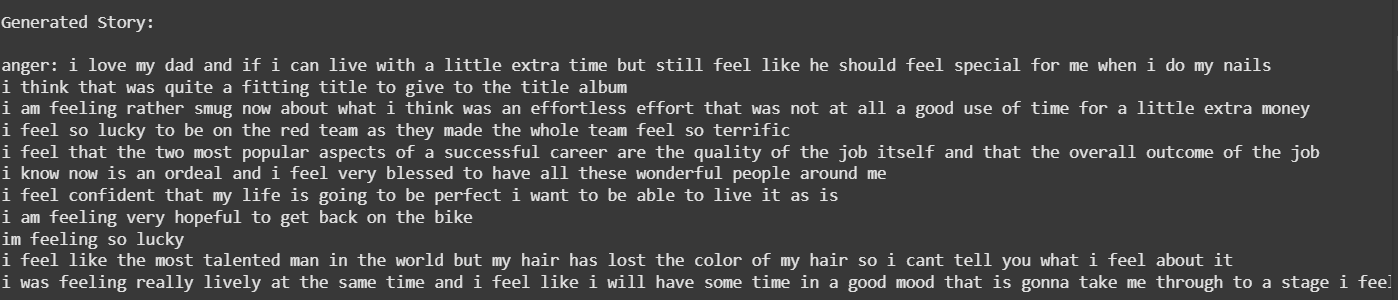
### BERTSCORE

The purpose of this method involves calculating semantic similarity through contextual embeddings derived from a pre-trained BERT model that studies textual content at a more profound level than simple words. What Happens: Input and Generation: The prompt generates a story.

* BERTScore Calculation: Through the bert\_score library this system calculates the cosine similarity value between context-aware embeddings which have been processed by a BERT model (e.g. roberta-large) for both the story text and reference text. The precision measurement evaluates which parts of the story content correspond with the reference content. The story contains what percentage of information present in the reference.
* F1: Harmonic mean of P and R. BERTScore evaluates semantic correlation between these concepts although the text uses different expressions to describe "hopeful girl" and "happiness".
* The F1-score produces an output value of 0.7500 which exceeds BLEU/ROUGE scores because BERTScore analyzes contextual meaning. The value range of 1 represents clear semantic similarity between two texts. BERTScore allows the story content to match specific semantic meaning of "joy" through its narrative structure (for instance a hopeful train station scene) while accepting major linguistic variations.

|  |  |  |
| --- | --- | --- |
| Metric | Score | |
| BLEU Score | 0.0051 | |
| ROUGE-1 | 0.0856 | |
| ROUGE-L | 0.0700 | |
| METEOR Score | 0.1919 | |
| BERTScore | 0.8149 | |
|  | |
| Sentiment | POSITIVE (score: 0.9948) | |



Sample1 genrated story.

Sample 2 genrated story.

## CONCLUSION

The story generation system utilizes fine-tuned GPT-2 model technology to create emotionally rich narratives which produce distinct outcomes from conventional algorithmic generation processes because it has a transformer-based structure that learns from contextual components. GPT-2 gathers semantic information through 12 transformer layers with multi-head self-attention that detects profound relationships of meaning and emotion embedded within input prompts ("joy: A girl waiting at the train station, full of hope"). Through its training on train.txt GPT-2 creates emotionally appropriate stories by choosing probable words from learned distributions which it steers with top-k, top-p and temperature settings thus producing creative texts such as "The sun shone brightly as she smiled." The traditional method of text generation produces formulaic and repetitive outcomes because GPT-2 utilizes its 124 million parameters with pre-training capabilities to produce superior human-like results with emotional authenticity.