Twitter Dataset Analysis

Group 17

Name	**Student Number (SN)**	**Contribution**
Karan Goel	7836685	14.28%
Alvin Jose	8066358	14.28%
Ashutosh Bhosale	7795786	14.28%
Banin Sensha Shreshta	8447196	14.28%
Gaurav Adarsh Santosh	7032663	14.28%
Lino Thankachan	7926017	14.28%
Rishab Manokaran	7863974	14.28%

CSCI946 Big Data Analytics Assignment 2

September 20, 2024

1 Introduction

In this assignment, the primary objective is to detect misinformation on social networks by identifying profiles that are incorrectly classified as human or non-human. The focus of the analysis will be on the Twitter user dataset, where we aim to explore methods to distinguish between genuine and artificially generated accounts. To achieve this, we will follow a structured approach outlined in Tasks 1-4.

Task 1 involves designing a comprehensive big data analytics project, adhering to the principles of the Big Data Analytics Lifecycle.

In Task 2, the dataset will be processed by taking into account the various data types and properties. Core models and algorithms will be applied, including regression, association rules, clustering, classification, and text processing methods.

Task 3 focuses on visualizing the dataset and utilizing visual representations to evaluate the analysis results. This step will provide valuable insights into the data and help validate the findings.

Finally, Task 4 entails a detailed study of various profile factors, such as text, color, and tweet content. Based on this analysis, recommendations will be made to adjust the classification of human and non-human profiles.

2 Task 1: Data Analysis Design

2.1 Business Domain

Objective

The primary aim of this assignment is to identify and analyze profiles on social networks that are mistakenly recorded as human or non-human (e.g., bots). Specifically, the objective is to classify Twitter user profiles into human or non-human categories based on available metadata and text information.

Dataset

The dataset used for this classification task is the Twitter User Gender Classification dataset, which is available on Kaggle.

Resources & Goals

Data Source: The dataset consists of Twitter user profiles, which include tweets, descriptions, link color, sidebar color, and other metadata.
Primary Task: The task is to classify the gender of Twitter profiles into two categories: human or non-human.
Key Questions:

- How well do the words in tweets and profiles predict the user as human or non-human?
- What are the specific words that strongly predict human or non-human profiles?
- How well do other factors (like link color, tweet count, retweet count, favourite number) predict whether a profile is human or non-human?

2.2 Framing the Problem & Initial Hypotheses

Problem Type

This is a supervised learning problem where the task is to predict a categorical variable (gender: male, female, or brand).

Hypotheses

- **Null Hypothesis (H0)**: Words in tweets and profiles do not have a significant effect on predicting whether a user is human or non-human (i.e., the predictive power is random or weak).
- **Alternative Hypothesis (H1)**: Words in tweets and profiles significantly affect the prediction of whether a user is human or non-human (i.e., they provide strong predictive power).
- **Additional Hypotheses**:
  - **H0**: Other factors (such as link color, tweet count, retweet count, favourite number) are not good predictors of whether a user is human or non-human.
  - **H1**: Other factors can strongly predict whether a user is human or non-human.

2.3 Data Preparation

In the data preparation process, several important preprocessing steps were applied to ensure consistency and effective representation of text data. These steps included:

- **Stemming & Lemmatization**: Words were reduced to their base or root forms to avoid variations in word forms affecting the analysis.
- **Lowercasing**: Text was converted to lowercase to eliminate discrepancies between uppercase and lowercase letters.
- **Tokenization**: The text was split into individual words or tokens.
- **Bag of Words**: A simple yet effective method for text representation was used to analyze the frequency of words in the dataset.
- **Word2Vec Model**: This model was applied to generate word embeddings for further analysis.

2.4 Visualizing Data

During the data visualization process, several techniques were employed to gain insights from the text data. A **word cloud** was generated to display the most frequent words associated with different gender categories (male, female, and brand). Additionally, **word distance and similarity** techniques, such as cosine similarity, were applied to understand which words or phrases were more closely related to each gender category. These visualization methods provided an intuitive understanding of the text data.

2.5 Model Selection

In the model selection phase, various machine learning models were explored to address classification, clustering, and regression tasks:

Classification Models

Models such as **K-Nearest Neighbors (KNN)**, **Support Vector Machines (SVM)**, **Decision Trees**, and **Random Forests** were considered. These models are effective for categorizing data based on the patterns and relationships within the dataset.

Clustering Models

For clustering tasks, models like **K-Means**, **DBSCAN**, and **Self-Organizing Maps (SOM)** were employed. Clustering models are helpful in discovering natural groupings in the data, allowing for an unsupervised learning approach to identify hidden patterns without predefined labels.

Regression Models

Both **Linear Regression** and **Logistic Regression** were used. Logistic Regression, in particular, was useful for binary classification tasks such as predicting whether a profile is human or non-human.

Neural Networks

Neural networks were also utilized to build deep learning models for more advanced classification tasks.

2.6 Training and Testing

Cross-Validation

To prevent overfitting and assess the generalization of the models, **k-fold cross-validation** was applied. This technique divides the dataset into multiple folds, allowing each fold to be used for both training and validation, resulting in a more robust evaluation of model performance.

Grid Search

Grid search was employed to optimize hyperparameters by systematically exploring different model configurations. This approach enabled the selection of the best combination of parameters for optimal model performance.

2.7 Final Deliverables

For evaluating the models, the following metrics were used:

- **Confusion Matrix**: This was used to visualize the performance of each classification model.
- **Accuracy**: The percentage of correctly classified instances.
- **Precision**: The proportion of true positives out of all positive predictions.
- **Recall**: The proportion of true positives out of all actual positives.
- **Loss Curves**: For neural networks, loss curves were used to visualize the training process and detect overfitting or underfitting.

3 Task 2: Data Processing & Model Application

The dataset was loaded using the `load_data` method from the `Task2` class, which reads a CSV file containing Twitter user data. After loading, the data was preprocessed to make it suitable for the classification task.

3.1 Data Processing

Below are the details of the steps undertaken in the Data Processing phase.

Preprocessing

- Selected rows with a `gender:confidence` score greater than 0.9 for data quality.
- Restricted `gender` labels to three categories: `male`, `female`, and `brand`.
- Relabeled `gender`: grouped `male` and `female` into a `human` class, and `brand` into a `non-human` class.
- Filled missing values with empty strings (`''`) to prevent issues during processing.

Feature Selection

Key features selected for classification include `gender`, `fav_number`, `retweet_count`, `tweet_count`, and `text`. These features were processed with `StandardScaler` for numerical attributes and `LabelEncoder` for categorical variables.

Feature Extraction

Two techniques were employed to convert processed text data into a numerical format:

Bag of Words (BoW)

Text data was vectorized using `CountVectorizer`, resulting in a sparse matrix of token counts, stored as `bow_feature` in the dataset (see Figure 4).

Word2Vec Embeddings

Word2Vec embeddings were created using the Gensim library, capturing semantic information. The average embedding for each tweet was calculated and stored as `word2vec_embeddings` in the dataset.

3.2 Models Application

Various methods, including regression, association rules, clustering, classification, and text processing, were considered for this study. Each method offers unique advantages and is suited for different types of data and problems. Below are the details of each method applied.

Clustering Models

K-Means Clustering

K-Means was applied with $K = 2$ to distinguish between two major groups, hypothesized to be human vs. non-human profiles. The model was trained on the training set and evaluated using the Silhouette Score, a metric that assesses how well the clusters are separated.

Results: The Silhouette Score was 0.366, indicating moderate clustering performance. This suggests that while K-Means could form distinguishable clusters, there was some overlap, likely due to the noisy nature of social media data.

DBSCAN (Density-Based Spatial Clustering)

DBSCAN, a density-based clustering algorithm, was applied with eps = 0.5 and min_samples = 5. This method is particularly suitable for identifying arbitrary-shaped clusters and noise (outliers).

Results: The Silhouette Score was 0.9949, showing highly effective clustering. The high score indicates that DBSCAN was able to separate the profiles into distinct groups with minimal overlap. This is particularly useful for the task at hand, as DBSCAN can also flag outliers that don't fit into either human or non-human profiles.

Self-Organizing Map (SOM)

A 5x5 SOM grid was used to train the model for 200 iterations. SOMs map high-dimensional data to a two-dimensional grid, allowing for easy visualization of clusters.

Results:

- The Quantization Error was 2.97, indicating the average distance between each profile and its closest node in the SOM. A lower QE would be preferable, but this value shows that the SOM performed reasonably well.
- The model identified 25 unique nodes (clusters), suggesting that the SOM captured more granular patterns within the profiles, potentially distinguishing between different types of human or non-human users (e.g., highly active users, bots, brands, etc.).

Classification Models

K-Nearest Neighbors (KNN)

KNN is a simple, non-parametric classification method that assigns a label based on the majority vote among its nearest neighbors. We utilized the Euclidean distance metric, and the optimal number of neighbors (k) was determined using GridSearchCV. KNN effectively handles high-dimensional datasets, such as Twitter profiles, where it can identify subtle distinctions between human and non-human behavior.

Results:

Metric	**Value**
Best KNN Params	{'model__n_neighbors': 3, 'model__weights': 'uniform'}
Accuracy	0.72
F1 Score	0.612
Recall	0.72

Table 1: K-Nearest Neighbors Results

Support Vector Classifier (SVC)

SVC constructs a hyperplane in high-dimensional space to separate classes. The radial basis function (RBF) kernel was employed, and the regularization parameter (C) was optimized through cross-validation. SVC is

well-suited for datasets with complex boundaries, excelling in high-dimensional feature spaces typical in social media networks.

Results:

Metric	Value
Best SVM Params	{'model__C': 10, 'model__kernel': 'rbf'}
Accuracy	0.725
F1 Score	0.651
Recall	0.725

Table 2: Support Vector Classifier Results

Decision Tree Classifier

Decision Trees partition data based on feature values to maximize information gain. We experimented with various tree depths to mitigate overfitting. While Decision Trees are interpretable and compatible with both categorical and numerical data, they can exhibit high variance if not carefully tuned.

Results:

Metric	Value
Best Decision Tree Params	{'model__max_depth': 3, 'model__min_samples_leaf': 2, 'model__min_samples_split': 2}
Accuracy	0.755
F1 Score	0.719
Recall	0.755

Table 3: Decision Tree Classifier Results

Random Forest Classifier

Random Forests combine multiple decision trees trained on different data subsets, reducing variance through averaging. We optimized the number of trees and maximum depth for improved performance. This ensemble method is effective in noisy datasets and helps identify feature importance, revealing which aspects of Twitter profiles (e.g., tweets, location) are most predictive of bot behavior.

Results:

Metric	Value
Best Random Forest Params	{'model__max_depth': 7, 'model__min_samples_leaf': 2, 'model__min_samples_split': 2
Accuracy	0.72
F1 Score	0.620
Recall	0.72

Table 4: Random Forest Classifier Results

Regression Models

Linear Regression

Linear regression is a fundamental statistical method used for predicting a target variable by fitting a linear relationship between the target and one or more predictor variables. We applied ordinary least squares (OLS) regression to the dataset, assessing the model's performance using metrics like R-squared and mean squared error.

Results:

Classification Report (Linear Regression used for Classification):

	precision	recall	f1-score	support
0	0.77	0.66	0.71	142
1	0.38	0.52	0.44	58
accuracy			0.62	200
macro avg	0.58	0.59	0.58	200
weighted avg	0.66	0.62	0.63	200

Logistic Regression

Logistic regression is a statistical method used for binary classification problems, where the goal is to model the probability that a given input belongs to a particular class. We employed a logistic regression model with a binary output (human and non-human), using the logistic (sigmoid) function to map predicted values to probabilities.

Classification Report (Logistic Regression with SMOTE):

	precision	recall	f1-score	support
0	0.75	0.67	0.71	142
1	0.36	0.47	0.41	58
accuracy			0.61	200
macro avg	0.56	0.57	0.56	200
weighted avg	0.64	0.61	0.62	200

Neural Network

Neural networks are a powerful class of models that can capture complex patterns in data through multiple layers of interconnected nodes. We implemented a feedforward neural network architecture, using the ReLU activation function for hidden layers and softmax for the output layer. The model was trained using backpropagation and optimized with the Adam optimizer.

Results:

Metric	Value
Accuracy	0.8262
Loss	0.3760
F1 Score	0.71

Table 5: Neural Network Results

4 Task 3: Visualization

4.1 Data Visualization

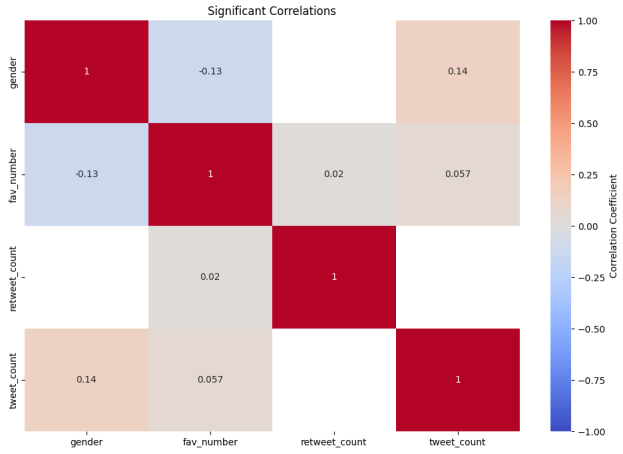

Figure 1: Correlation Matrix

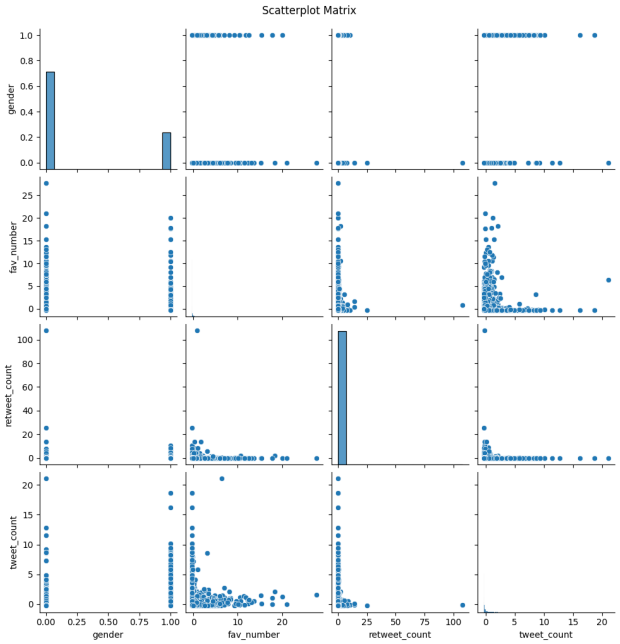

Figure 2: Pair Plot

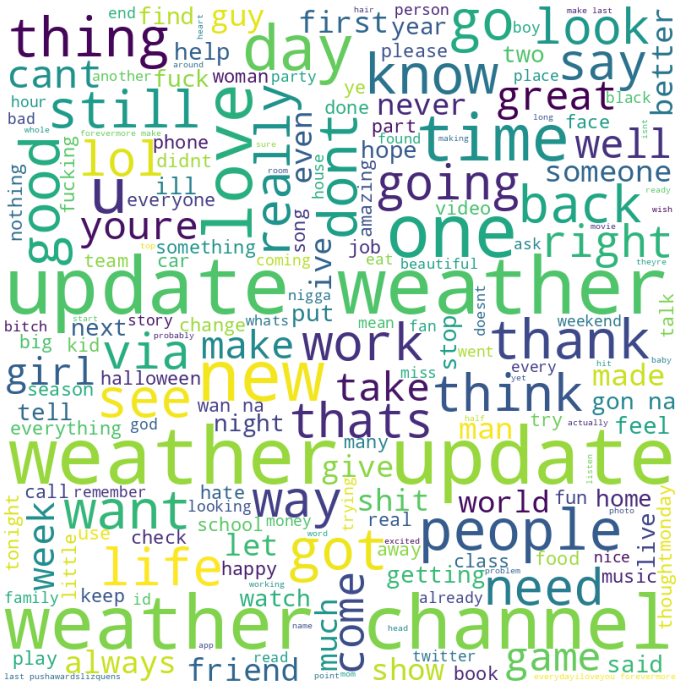

Figure 3: Word Cloud

	gender	fav_number	retweet_count	text	tweet_count	bow_feature	word2vec_embeddings
0	0	-0.348345	-0.047969	robbie e responds critic eddie edward worldit...	0.582729	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...	[0.04739282, -0.010395167, 0.015648082, -0.004...
1	0	-0.342754	-0.047969	felt friend living story retired ian1	-0.244501	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...	[0.2457285, -0.064767666, 0.07366451, -0.03446...
3	0	-0.331736	-0.047969	hi looking url use dont typically see advanced...	-0.290685	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...	[0.4261804, -0.12715583, 0.11448964, -0.063192...
4	0	2.719920	-0.047969	watching neighbour sky catching neighbs xxx xxx	-0.052738	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...	[0.07816511, -0.020627912, 0.027190495, -0.005...
5	0	-0.027607	-0.047969	ive seen people train lamp chair tv etc	-0.144067	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...	[0.36296287, -0.10343567, 0.10371529, -0.05848...

Figure 4: Data Frame

4.2 Clustering Visualization

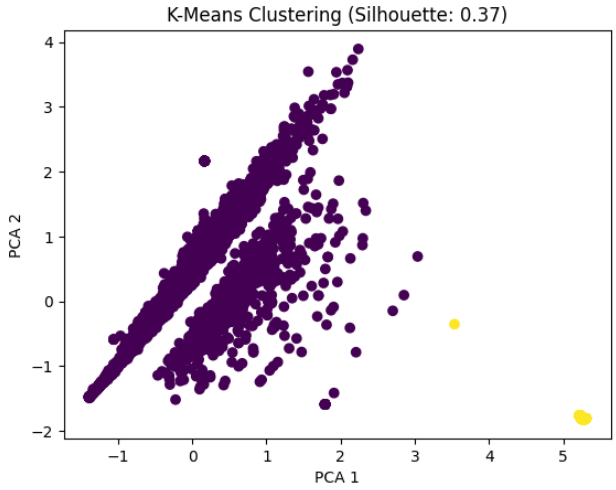

Figure 5: K Means

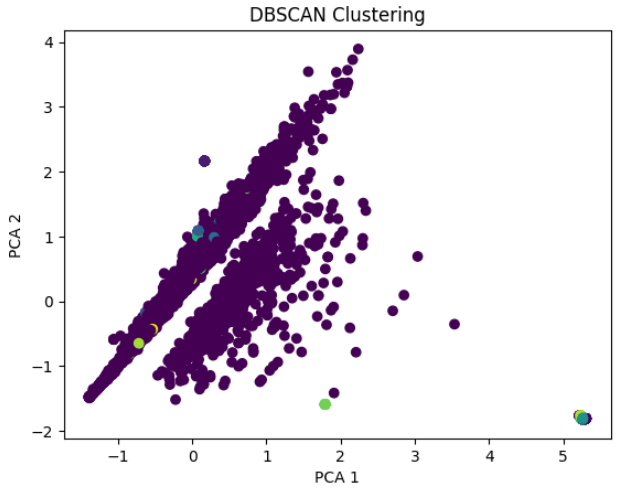

Figure 6: DB Scan

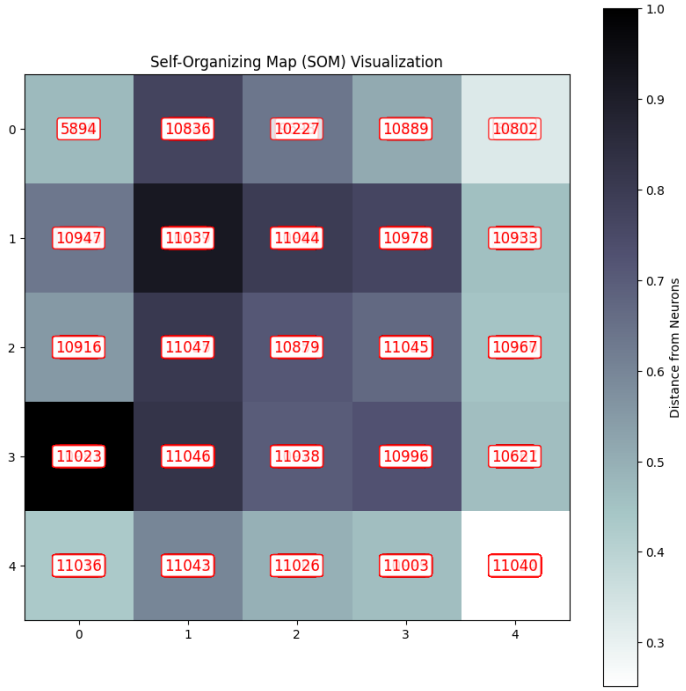

Figure 7: Self Organizing Maps

4.3 Neural Network Visualization

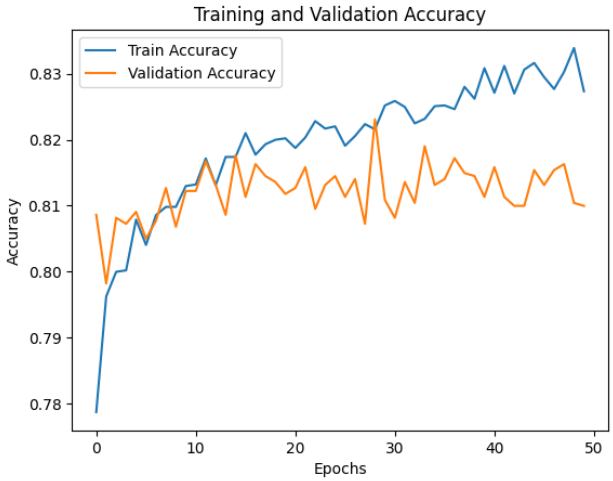

Figure 8: Accuracy Curves

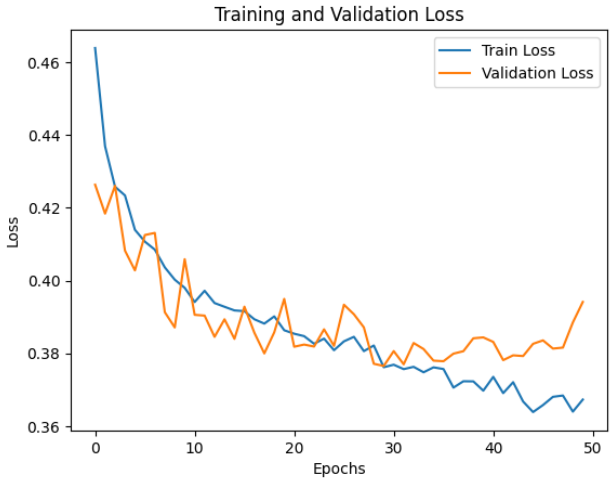

Figure 9: Loss Curves

5 Task 4: Suggestion to Amend Human and Non Human Profile

In this section, we analyze various profile factors that could influence the classification of Twitter profiles as human or non-human. The factors examined include text content, profile colors, tweet engagement metrics, and overall user behavior.

5.1 Text Content Analysis

The analysis of tweet content focused on identifying key terms and phrases that differentiate human users from non-human accounts. We utilized techniques such as Word Cloud and frequency analysis to gain insights into the language used by different profile types.

Results:

- Human profiles displayed a higher frequency of positive sentiments, indicating more personal and engaging interactions.
- Non-human profiles were often characterized by neutral sentiments, reflecting a lack of emotional engagement.

Keyword Frequency

We conducted a keyword frequency analysis to identify specific terms that were prevalent in each category. Using the Bag of Words model, we generated frequency distributions for each gender class.

Results:

- Terms such as "love," "happy," and "community" were predominantly found in human profiles.
- Non-human profiles frequently included keywords like "update," "channel" etc indicating promotional content.

5.2 Profile Color Analysis

While we did not identify significant differences among the profile colors, the performance of the regression model improved with the incorporation of these features, suggesting a potential link between these factors.

5.3 Engagement Metrics

Engagement metrics such as retweet counts, tweet counts, and favorites were analyzed to evaluate the activity levels of human versus non-human profiles.

Results:

- Human accounts tended to have higher retweet and favorite counts, indicative of active participation in conversations.
- Non-human profiles, while they may have high tweet counts, often lacked engagement, demonstrating lower retweet and favorite ratios.

5.4 Recommendations for Classification Adjustments

Based on the analysis of profile factors, we recommend the following adjustments to improve the classification of human and non-human profiles:

- Incorporate keyword frequency metrics as additional features in the classification models to enhance predictive accuracy.
- Utilize color analysis as a supplementary criterion for distinguishing between human and non-human profiles.
- Monitoring engagement metrics closely can serve as key indicators of user authenticity.

References

Bishop, C. M. (2006). *Pattern recognition and machine learning.* Springer. Retrieved from https://www.springer.com/gp/book/9780387310732

Developers, N. (2024). *Numpy: The fundamental package for scientific computing with python.* Retrieved from https://numpy.org

Géron, A. (2019). *Hands-on machine learning with scikit-learn, keras, and tensorflow.* O'Reilly Media. Retrieved from https://www.oreilly.com/library/view/hands-on-machine/9781492032632/

learn Developers, S. (2024). Scikit-learn user guide. *Scikit-learn Documentation.* Retrieved from https://scikit-learn.org/stable/user_guide.html

Team, P. D. (2024). *Pandas: Python data analysis library.* Retrieved from https://pandas.pydata.org

VanderPlas, J. (2016). *Python data science handbook: Essential tools for working with data.* O'Reilly Media. Retrieved from https://jakevdp.github.io/PythonDataScienceHandbook/