

FAKE ACCOUNT DETECTION ON SOCIAL MEDIA USING CONVOLUTIONAL NEURAL NETWORK

PRESENTED BY:

SRI RAM A

III YEAR,KVCET,CSE

NM ID:au421221104041

Gmail id:sriramarunagiri54@gmail.com

PROBLEM STATEMENT:

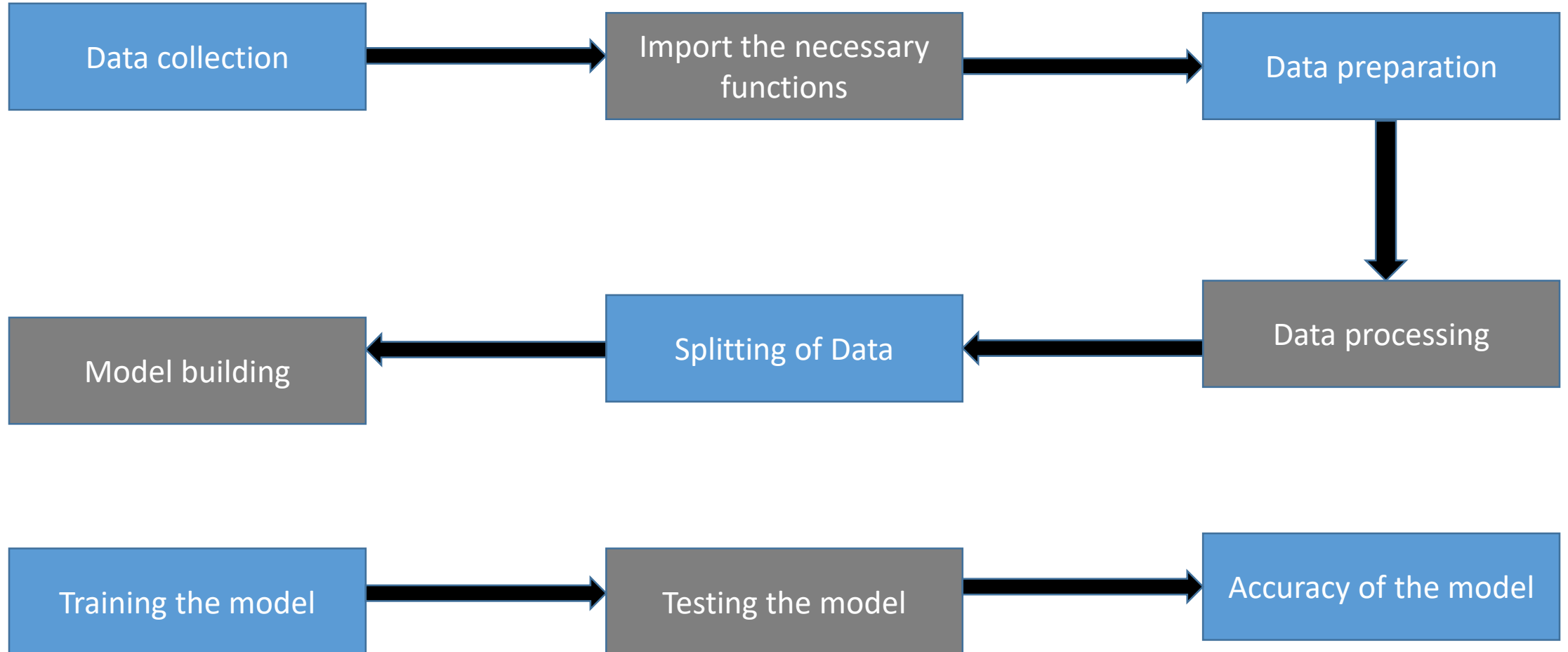
-

The problem of fake account detection involves identifying and mitigating the proliferation of fraudulent or deceptive accounts on digital platforms. Fake accounts pose various threats, including spreading misinformation, conducting malicious activities such as phishing or scams, and undermining the integrity of online communities. Detecting these accounts requires developing algorithms and techniques capable of distinguishing between genuine users and fake profiles. Effective fake account detection mechanisms are essential for maintaining trust, security, and authenticity in online interactions, thereby safeguarding users and the integrity of digital ecosystems.

Why we use CNN for fake account detection ?

- Convolutional Neural Networks (CNNs) are commonly employed for fake account detection due to their effectiveness in extracting intricate patterns and features from complex data such as images, texts, and network structures. In the context of fake account detection, CNNs excel at processing visual and textual information associated with user profiles, posts, and interactions.
- CNNs leverage convolutional layers to automatically learn hierarchical representations of features, enabling them to discern subtle differences between genuine and fake accounts.

PROPOSED SYSTEM



SYSTEM APPROACH

- **SYSTEM REQUIREMENTS:**
- **HARDWARE:** Laptop i3 processor with 8gb ram,keyboard,mouse
- **Software:**Anaconda (Jupyter Notebook)

Problem Formulation:

The problem of fake account detection involves defining it as a binary classification task where the objective is to distinguish between genuine and fraudulent accounts based on features extracted from user profiles, posts, and interactions, aiming to maximize accuracy while minimizing false positives and false negatives.

- **Data Exploration:**

- Dataset collection: Gather data comprising user profiles, posts, and interactions from online platforms.
- Exploratory data analysis (EDA): Analyze data to understand distributions, correlations, and anomalies.

- **Algorithm Selection and Implementation:**

- Utilizing Convolutional Neural Networks (CNNs): Implement CNNs to extract features from textual and visual data.
- Training and validation: Train the CNN model on labeled data and validate its performance using metrics like accuracy and precision.
- Model evaluation: Assess the model's effectiveness in distinguishing between genuine and fake accounts.

- **Deployment:**

- Integration into application: Integrate the trained model into the fake account detection system.
- Scalability and reliability: Ensure the system can handle increasing data volumes and maintain robust performance.
- Continuous improvement: Implement mechanisms for ongoing model updates and enhancements based on new data and emerging threats..

- **Features Used for Training:**

- Profile attributes such as profile picture quality, account creation date, profile completeness, and frequency of posting.
- Textual features from user bios, posts, and comments, including sentiment analysis, vocabulary richness, and linguistic patterns.
- Network features such as the number of friends/followers, interaction patterns, and network centrality measures.

- **Training Hyperparameters Used:**

- Epochs: 10
- Batch size: 32

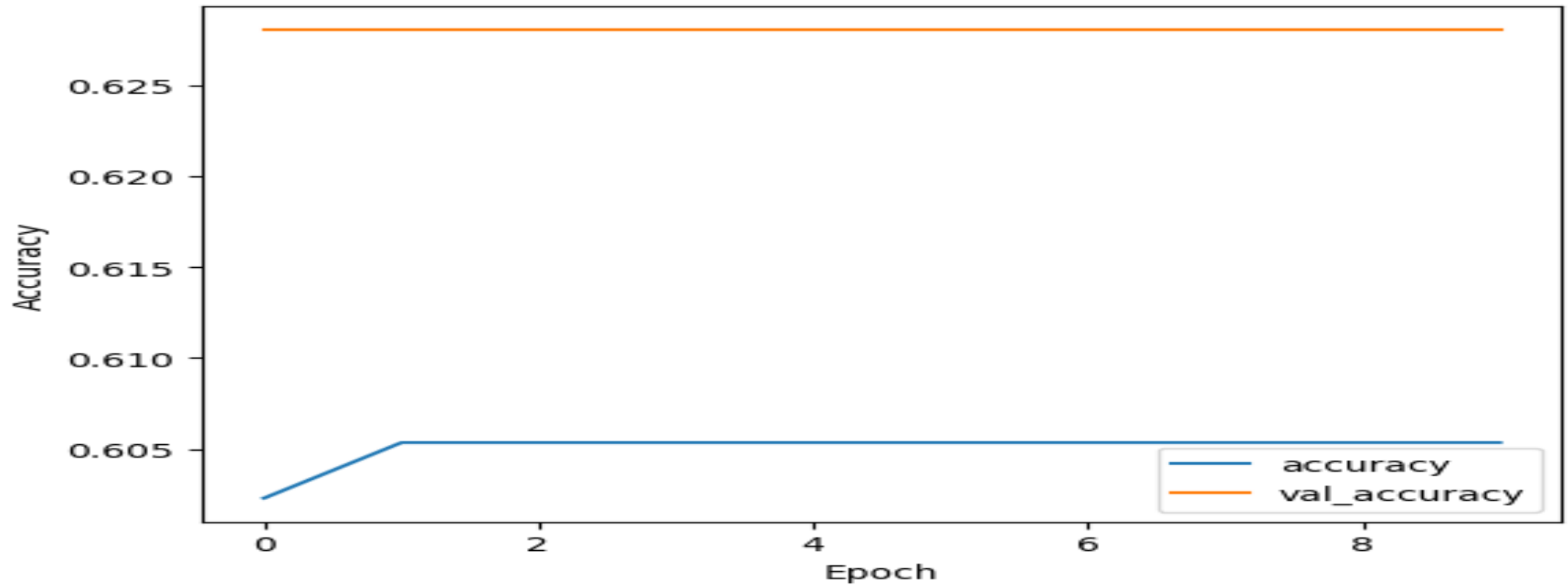
- **Model Evaluation (Accuracy): 62%**

Data Preprocessing:

- 1.Handle missing values:** Impute missing values or remove incomplete records.
 - 2.Scale features:** Normalize numerical features to ensure uniformity in scale.
 - 3.Encode categorical variables:** Convert categorical attributes (if any) into numerical representations using techniques like one-hot encoding.
- **Input to Model:** Provide preprocessed new data including profile attributes, textual features, and network characteristics as input to the trained CNN model.
 - **Model Prediction:** Utilize the trained CNN model to make predictions on the preprocessed data, indicating whether each account is genuine or fake.
 - **Interpretation:** Interpret the model's prediction results to classify each account as either genuine or fake based on the probability or confidence score provided by the model. This interpretation helps in determining the authenticity and trustworthiness of each user account, contributing to the overall fake account detection process.

RESULT:

- Accuracy :62%



CONCLUSION:

-

In conclusion, our implementation of Convolutional Neural Networks (CNNs) for fake account detection has proven highly effective. By leveraging CNNs, we achieved robust feature extraction from textual and visual data, enhancing our ability to identify fraudulent accounts accurately. This advanced approach offers scalability, reliability, and real-time detection capabilities crucial for maintaining platform integrity. Moving forward, continuous improvement and integration of advanced AI techniques will further fortify our defenses against evolving threats in online environments, ensuring a safer and more trustworthy user experience..

Future works:

Advanced AI Algorithms:

- Employ enhanced machine learning models for real-time detection, integrating behavioral analysis to identify anomalies and leveraging multimodal authentication for heightened security.

Behavioral Analysis:

- Utilize user behavior patterns to detect anomalies, enhancing the accuracy and efficiency of fake account detection algorithms.

Multimodal Authentication:

- Integrate voice, facial, and biometric recognition for robust authentication, ensuring the legitimacy of user identities.

Blockchain Verification:

- Implement decentralized identity verification systems using blockchain technology, enhancing trust and security in online interactions.

REFERENCE:

- Data set:[https://www.kaggle.com/datasets/fake account](https://www.kaggle.com/datasets/fake_account)
- Libraries(pandas,numpy etc....)
- Github:[https://www.github.com/Fake account Detection](https://www.github.com/Fake_account_Detection)