National College of Ireland

Real-Time Detection of Social Engineering Threats in Social Media Posts

MSc Research Project
MSc Cybersecurity

Jagadish Maranachakanahalli Dhananjaya
Student ID: X23103272

School of Computing
National College of Ireland

Supervisor: Liam McCabe

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name:	Jagadish Maranachakanahalli Dhananjaya
Student ID:	X23103272
Programme:	MSc In Cybersecurity **Year:** 2024
Module:	MSc Research Practicum
Supervisor:	Liam McCabe
Submission Due Date:	16/09/2024
Project Title:	Real-Time Detection of Social Engineering Threats in Social Media Posts.
Word Count:	7205 **Page Count** 23

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.
ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature: Jagadish Maranachakanahalli Dhananjaya

Date: 14/09/2024

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST

Attach a completed copy of this sheet to each project (including multiple copies)	□
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	□
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	□

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Table of Figures

Table of Acronyms

Acronym	Full Form
YOLO	You Only Look Once
NLP	Natural Language Processing
GPU	Graphics Processing Unit
CNN	Convolutional Neural Network
ATM	Automated Teller Machine
DOB	Date Of Birth

Real-Time Detection of Social Engineering Threats in Social Media Posts

Jagadish Maranachakanahalli Dhananjaya
X23103272

Abstract

Social engineering attacks utilize online information, such as data from social media platforms, to indirectly obtain personal information. This method poses significant security risks by exploiting publicly available data to piece together sensitive information, making it harder for individuals to recognize and prevent such attacks. This research shows how we can predict susceptibility of social engineering by analysing social media posts. This research utilizes the YOLO (You Only Look Once) model, a real-time object detection system that processes an entire image in a single pass by predicting bounding boxes and class probabilities simultaneously, to detect objects such as laptops and dogs. Additionally, natural language processing is employed to analyse text for information like dates of birth and names, which are utilized in social engineering attacks. The application combines this analysis to provide users with immediate warnings by alerting them to potential security threats before they post on social media. This integrated approach, using both visual and text data, enhances the ability to predict social engineering attacks. This research contributes to cybersecurity by offering a proactive tool for protecting personal and corporate information shared on social media.

1 Introduction

The dangers associated with the cyber-attacks are continually evolving by becoming more complex with the technological advancements. Social engineering attacks are particularly focused on exploiting online information provided by specific users to gain unauthorized access to information systems. These attacks exploit human vulnerabilities by making them challenging to identify and prevent. (Katharina Krombholz, 2015) in his research he explains the common types of social engineering attacks include phishing, pretexting, baiting, and tailgating. Many of these attacks use trust and information from social media to persuade individuals into disclosing sensitive information or taking actions that compromise their security. According to (Magazine, 2020) by the time of 2025, the global cost of cybercrime is estimated to reach $10.5 trillion, with social engineering attacks contributing significantly to this figure. This rise can be attributed to the increasing dependence on technology and the growing complexity of cybersecurity threats.

(Nelson Duarte, n.d.) study say that attackers utilize a online information, such as social media post, for their attacks because it is often simpler and more effective than breaching technical defenses. For example, predicting card PINs based on publicly shared information like dates of birth on social media can be relatively simple. This method is particularly effective when

attackers use publicly available information on social media to exploit such personal details. However, for those without expertise in social engineering, these attacks can be more challenging, as their success depends on both the attacker's skill and the victim's susceptibility to deception.

(Sennovate, 2023) study say social media causes this risk by pushing users to publish excessive amounts of information which further allows hackers to launch their highly targeted attacks based on user post. These attacks can take many forms, including account takeovers, predicting ATM pins, answering security questions, all targeted at extracting critical information from social media posts. Maintaining the security of an organization and its workers social media activity is difficult due to the frequent sharing of information on these platforms. As a result, it is important to develop an effective method for detecting and preventing these attacks to minimize the financial and reputational harm that can occur.

Research Question

Can integrating the YOLO model with natural language processing for content analysis provide an effective early warning system to safeguard social media platform users against social engineering attacks?

Research Objectives

This research's main goal is to create an integrated system that uses image processing method to anticipate and prevent social engineering attacks.

This study aims to design and implement a predictive program which is capable of automated detection of social engineering attacks. The system will integrate with the elements of machine learning with the YOLO (You Only Look Once) object detection framework. This framework will enable the real time identification of visual cues of potential social engineering activities such as unauthorized access to confidential information which is displayed on screen.

The research will also incorporate with the NLP (Natural Language Processing) techniques to analyze the textual data for patterns which are commonly associated with the social engineering attacks. This includes detecting personal identifiers, sensitive information on text.

2 Related Work

This section illustrates the research in the context of academic literature by critically reviewing the major contributions in the field of social engineering attack detection.

2.1 Text Analysis in Social Engineering Detection

(Michael Fire, 2014) in his work on detecting social engineering attacks through user posts on social networks, Michael Fire utilized the natural language processing (NLP) techniques to identify the signs of phishing and scam attempts. Fire's approach involves in analysing a

language patterns, such as word frequency and sentiment, to detect potentially malicious messages. By integrating with the various features, including textual metadata and user behaviour, Fire's model improves the accuracy of detecting social engineering attacks. However, this study is limited by its focus on textual data which ignores the visual content that is frequently present in social media posts. This limitation affects the system's effectiveness in situations whenever critical information is encoded in the images rather than text, which has become typical in modern social media contexts.

(Tom N. Jagatic, 2007) in his study titled "Social Network Analysis for Phishing Detection," analyzes the communication patterns to show how vulnerable is social network users are to phishing attacks. He clarifies that targeted phishing attempts, created using the information from social media account, are significantly more effective than generic phishing emails. Jagatic's study highlights the importance of context and personalization in social engineering attacks. Through real-world experiments involving email phishing attacks, he shows a high success rate for attacks leveraging on social media content. However, Jagatic's study lacks an accurate mechanism for real-time detection and prevention, which is critical for mitigating such attacks. His approach focuses on illustrating users' vulnerabilities rather than giving a comprehensive solution for protecting them.

(Zeadally, 2015) In his research paper "Identity Deception on Social Networking Platforms: A Comprehensive Framework," Zeadally proposes an detailed framework for understanding and addressing the identity deception on social networking platforms. He categorizes different types of deception, such as identity theft and impersonation, and examines both user and developer perspectives to propose various detection strategies. These strategies include using historical records, analyzing social connections, and implementing advanced text analysis techniques to verify the user identities. Zeadally also addresses the major challenges by implementing these strategies, such as the computational costs associated with real-time detection and the potential privacy concerns that arise when analyzing user data. Despite these challenges, Zeadally's work lays a solid foundation for further research which aimed at improving the speed and accuracy of detection systems by addressing the ethical concerns, and developing scalable solutions which is suitable for diverse social networking environments.

2.2 Image analysis

When it comes to predicting social engineering attacks, utilizing social media post analysis and number of related works offer a basis in understanding the issues and methods involved in it. (Bo Mei, 2018) study "Inference Attacks Using Convolutional Neural Networks in Social Networks," conducted a relevant study on utilizing images and attributes in social networks to execute inference attacks with using convolutional neural networks (CNNs). This approach improves prediction accuracy by combining visual data with user attributes and it presents a defense mechanism that uses differential privacy to mitigate such threats. This method's strength is its high accuracy and innovative usage of neural networks, unlike classic algorithms such as decision trees, Naïve Bayes, and k-nearest neighbors (k-NN). However, there is a significant limitation that is complexity and processing intensity of CNNs, which may limit their practical implementation in real-time applications.

The paper “You Only Look Once: Unified, Real-Time Object Detection” by (Joseph Redmon, 2016) introduces a YOLO a unique approach to object detection that reframes detection as an single regression problem by predicting bounding boxes and class probabilities directly from the entire images via a single neural network. YOLO strength is its speed which can process images in real time at 45 frames per second with high mean average precision. However, it has limitations such as producing more localization errors and having difficulty with small objects in groups or uncommon aspect ratios due to the grid cell structure. However, YOLO single architecture reduces the detection pipeline and enhances the performance. But there requires a further improvement is needed to improve localization accuracy and handle different object configurations effectively.

3 Research Methodology

This research methodology aims to develop a robust web application designed to predict social engineering attempts in real-time as users share content on social media. The application will employ advanced image processing using the (Wei Fang, 2019) YOLO (You Only Look Once) object detection framework and (Dhruvi D. Gosai, 2018) natural language processing (NLP) techniques. This web application will identify the any disclosures of confidential while sharing on social media that user may post without intention or aware of the situation. Social engineering attacks can exploit shared information, such as birthdays or names, to guess ATM PINs or answer security questions. This motivates the development of a website that analyses text and images in social media posts to alert users and reduce the risk of disclosing sensitive information.

Equipment and Software used

To develop the social engineering predictor a high-performance GPU was used to detect real time image processing tasks. The software uses Python for its extensive libraries and ease of use. Here OpenCV was chosen for its powerful image processing function and the flask framework was utilized to build the web application interface. PyCharm integrated development environment (IDE) was used for coding and debugging (JetBrains, n.d.).

Data Collection and Analysis

The raw data for this research is gathered from the user inputs where user can submit their personal details and upload images in the social media. When an image is uploaded, it will then be pre-processed to create a blob suitable for the YOLO model, which will detect the objects such as dogs, cars number plate and laptops screen etc. These detected objects are then analysed to identify the potential security threats such as the presence of confidential information. In addition to that, the text data from user inputs is analysed to extract the potential personal information that could be used for social engineering attacks such as date of births and names related to security questions. This both result from both image and text analyses are then combined and presented back to the user as an alert.

Scenario Set-Up and Case Studies

Scenario 1: Object Detection in Uploaded Images

In this first scenario firstly, users were asked to upload an image which contain common objects such as dogs, cars, laptops etc. The main aim was to evaluate the YOLO model ability to detect the objects in the images. User also provided the personal details along with images to stimulate a typical use case where both image and text data are analysed together. After detected the objects in the image the web application generated security insights that were customized to the specific risks associated with every single object category. For example, if a laptop was spotted in the object a warning message will be displayed has there may contain a potential threat while sharing this post.

Scenario 2: Text Analysis for Security sensitive Information

The second scenario is focused on text analysis capabilities of the application. For this user enter their personal details that include security sensitive information such as date of birth, family members names, dog name. The goal was to test the natural language processing techniques which is used to extract and analyse this information. After it thoroughly analysed the text it alerts the user for example, if a user posted a DOB (DD/MM/YYY) of his child, the application tries multiple combinations (DDMM, MMDD, MMYY, YYDD) of ATM pin numbers in the DOB and display the alert message to user.

4 Design Specification

In design specification of the social engineering attack predictor application, we have used two method advanced image processing and natural language processing (NLP) techniques to identify the security threat which is uploaded from user under the image and text details. This architecture is divided into client and server mode where both has different roles and responsibilities to effectively analysis the result.

4.1 Techniques and Architecture

4.1.1 YOLO Object Detection

This research uses the YOLO (You Only Look Once) for object detection which is integrated with the OpenCV. This YOLO model is loaded with pre trained weights, configuration file and class names. The process of detecting objects in image using YOLO involves several steps.

4.1.1.1 Loading the model

- Pre trained weights: This model uses the pre trained weights which is obtain by training the YOLOv3 network on a large dataset which contain different types of objects (Kaggle, n.d.).

- Configuration file: This configuration file (yolov3.conf) contains the architecture details of the YOLOv3 model by specifying the layers, filters and other hyperparameters which are used during the training process (Kaggle, n.d.).
- Class Name: This class names (coco.names) contain the list of object classes which YOLOv3 model can detect, Such as dog, car and laptop (Kaggle, n.d.).

4.1.1.2 Preprocessing the input image

- After user inputted the image, it converts the image which is suitable for the YOLOv3 model. This involves resizing the image to the dimensions which is expected by the model typically 416 * 416 pixels.
- After resizing the image, a blob is created from the image using OpenCV. This blob is a 4-dimensional binary large object which normalizes the pixel values and prepare the image for neural network input.

4.1.1.3 Forward Pass through the Network

- A forward pass is performed through the network while using the net.forward function. This function processes the image using multiple convolutional layers to detect the objects in the image. Here the output is a set of predictions where each contain a bounding box, class probabilities and confidence scores.

4.1.1.4 Filtering Predictions

- The predictions from the YOLOv3 model include several bounding boxes with the associated confidence score and class probabilities. To filter these predictions the system selects only those with a confidence score above a certain threshold that is 0.5. This step ensures that only the most accurate detections are considered.
- For each object detection the image calculates the coordinates of the bounding box in the context of the original image size.

4.1.1.5 Non-Maximum Suppression (NMS)

- To remove the unimportant bounding boxes that overlap significantly this system uses Non-Maximum Suppression. NMS keep only the bounding box which has the highest confidence score among the overlapping boxes by ensuring a single detection per object.

4.1.1.6 Identifying Objects

- The filtered and suppressed bounding boxes are then analysed to identify the object detected in the image. Here each bounding box is associated with a class label such as dog, car, laptop and a confidence score.
- After that the web application generates a warning based on the identified objects. For example, if a laptop or computer screen is detected in the image the application warn the user about potential threat on the screen.

4.1.2 Natural Language Processing (NLP) Based Text Analysis

The web application uses the regular expression in python that is (re module) to analyse the text information which is provided by the user while posting in social media. This application focus on extracting the sensitive information such as date of birth and pet names and other personal details which can be used by an attacker to take control over their account. By extracting the dates formatted as DD/MM/YYY and generate a potential PIN combinations based on the date component. And, by identifying the names which is associated with the specific keywords example dog name, mother name to alert the users about the risk of sharing their personal information.

4.2 Software Requirements

- Python 3.12: Python is the programming language used for the entire project. Python 3.12 is chosen for its simplicity, readability and library support. Python provides the necessary environment for developing the web application, performing image processing and executing the NLP tasks.

- Flask: Flask is a micro web application framework for python which is used to build the web server. Flask is compact and adaptable, which makes it more suitable for creating the backend of our web application. (Index, n.d.) It handles the HTTP requests and sends them to the proper functions and generates the HTML templates. Flask will also additionally provide an easy integration with the project's various libraries and tools.

- NumPy: NumPy is an important Python scientific computing tool. Where it supports massive, multidimensional arrays and matrices as well as a wide range of high-level mathematical functions for manipulating the arrays. In this research (Index, n.d.) NumPy is used for the image processing task such as creating blobs from images, manipulating arrays and performing the mathematical computations which required for object detection.

- OpenCV: OpenCV (Open-Source Computer Vision Library) is an open-source computer vision and machine learning library. (Index, n.d.) The OpenCV contains the more than 2500 optimized algorithm where it can be used for various functions such as detecting and recognizing faces, identifying objects, classifying human actions in videos, tracking camera movements, extracting 3D models of objects, producing 3D point clouds from stereo cameras. In this project OpenCv is mainly used for loading the YOLO model for preprocessing images and performing the object detection.

4.3 Integrated Development Environment (IDE)

- PyCharm 2024.1.4: PyCharm is an integrated development environment used for computer programming which is mainly used for python (JetBrains, n.d.). Pycharm is vital for the creation of social engineering attack prediction system. PyCharm is a code

editor which included a feature such as code completion and real time mistake detection, and it make coding more efficient. The IDE has a strong debugging capability which allow step by step execution, variable inspection and runtime evaluation which were important for identifying and resolving the issues in the complex functions. Also this IDE support for web development which include HTML, CSS and javascript to create a responsive and user friendly interface.

4.4 Hardware Requirements

- Local Machine: This local machine is important for hosting and running the python flask application. This machine will handle the backend process by managing the HTTP requests and serve the HTML content.

- CPU: A multi core processor is recommended to efficiently handle the concurrent tasks and improve the overall performance. Example Intel Core i5 or AMD Ryzen 5

- Clock Speed: A higher clock speed is required (2.8 GHz or higher) which will enhance the processing speed.

- RAM: A minimum of 8 GB Ram is required to support the application and manage the multiple process simultaneously. However, 16 GB or more is recommended for better performance especially when dealing with the large images or data processing.

4.5 Functionality Description

- YOLOv3 Object Detection Model: This YOLO model is designed for efficient and accurate real time object detection. It processes the entire image in one forward pass making it more predictions for bounding boxes and class probabilities simultaneously. This model is used to detect specific objects in user uploaded images, and it generates warnings about security threats.

- NLP Based Text Analysis: This text analysis component uses a regular expression to identify and extract the sensitive information from the user post in social media. By recognizing the patterns that match dates and keywords, after that the application generates warnings about security threats such as uses of personal information in security question or as a PIN combination.

5 Implementation

The final stage of the implementation for the social engineering attack prediction application involves integrating with the several components to produce a working web application for capable of analysing text and images for security threats. The implementation process produces the multiple outputs, including the developed code and fully operational web interface.

5.1 Outputs Produced

5.1.1 Transformed Data

- Image Analysis Data: The web application analyses the uploaded images to detect specific objects such as dogs, cars and laptops by using the YOLO object detection model. The detection results include the identified objects within the image. These results are used to generate security related warnings such as the display of a laptop by indicating there is a potential threat in the image.

- Text Analysis Data: The web application analyses the user entered text data on web application to extract the sensitive data or information such as, date of birth or names which indirectly associated with the security questions. The analysis identifies the potential security risks such as use of date of birth for ATM pin or exposure of personal data that could be used in social engineering attacks.

5.2 Code Written

5.2.1 Backend

The backend of the web application was developed using the python and flask framework. Flask was chosen for its ease of integration with the other python libraries such as OpenCV and NumPY.

The server-side implementation involved creating a routes to handle HTTP requests such as managing the file uploads and process the data. In these two main methods were developed one for handling GET request to display the web interface and another one used for handling POST request to process the uploaded images and personal details.

The Flask application fetch the pre trained weights, configuration file, and class names to train the YOLO model. When an image was uploaded the server will preprocess the image by performing object detection using YOLO model and it apply non maximum suppression to filter the redundant detections. The results are then combined with the analysis of the user personal details to generate a comprehensive threat analysis which will further returned to the client-side display by warning the user.

5.2.2 Fronted

The fronted design is developed using HTML, CSS and Javascript to provide user friendly for uploading images and entering text details. The results of the analysis are displayed on this page.

HTML5: In the process of developing the web application I used HTML5 (Hyper Text Markup Language) for structuring the content of the web application. The HTML file contains a form for users to enter the caption and upload the images to post. It also contains placeholders for displaying the analysis results. HTML provides the backbone for the web application layout and content organization.

CSS (Cascading Style Sheets): CSS is used for styling HTML content. It also ensures that the web application has good visual appeal and user-friendly interaction. CSS rules are used to style the elements such as form, buttons and text areas. CSS enhances the overall user experiences by providing an attractive interface.

JavaScript: JavaScript is used for client-side scripting to handle the user interactions and dynamically updating the web page without reloading the page. Javascript functions handle from submission of send requests to the flask backend and then it process the responses to the user display of the analysis results.

5.3 Model Used

In developing the social engineering attack prediction web application, the existing YOLO model has been utilized for object detection. YOLO is an advanced real time object detection system this model operates under the principle of You Only Look Once which means it examines the entire image in one evaluation.

The YOLO model was integrated into the system using OpenCV module. This module load the pre trained weights (yolov3.weights), class name (coco.names) and a configuration file (yolov3.cfg) which are essential for the model operations. After receiving an uploaded image through the application interface, the system preprocesses the Image and send into YOLO model. During interface this model analyses the image and identifies the objects based on predefined class labels such as laptop, car and dog.

This YOLO object detection model plays an important role in enhancing the social engineering attack prediction. Its ability to accurately detect and analysis the uploaded images enables the system to provide security analysis for identifying the threat in the user post that could be used in social engineering attacks.

6 Evaluation

6.1 Experiment / Case Study 1: Predicting ATM PIN Numbers

This case study demonstrates the effective use of text analysis techniques to identify potential security risks associated with the sharing of personal details online like social media. The visual outputs provide a clear representation of the identified ATM PIN numbers derived from dates of birth, emphasizing the importance of privacy and security awareness in online activities.

Function Explanation

To analyse the text, I have implemented a function (analyze_text) to extract and analysing the potential ATM pin combinations from the DOB or important dates. The analyze_text function operates by first extracting the DOB using a regex pattern. Upon identifying a DOB in the DD/MM/YYYY format, it derives potential PIN combinations:

- Day and month (e.g., 2205 for 22/05/2000)
- Month and day (e.g., 0522 for 22/05/2000)

- Full year (e.g., 2000)
- Last two digits of the year combined with the day (e.g., 2022 for 2000 and 22 for December)
- Last two digits of the year combined with the month (e.g., 0005 for 2000 and 05 for December)

Expected Results:

- The function aims to identify potential ATM PIN combinations derived from a user's DOB found within the caption field.
- It will extract the DOB using the provided regex pattern (DD/MM/YYYY) and then generate all the listed PIN combination (day/month, month/day, full year, etc.) based on those digits.
- A pop-up window appears when a threat is detected. In that it must show generated PIN combinations and how they could be misused. And in the same pop-up window it should contain two options yes or no for posting the image or text on social media.
- If the user chooses yes for confirming the post, the application displays the post with a red border.
- Ideally, this would flag potential threats if someone might attempt to guess the user's ATM PIN using information gleaned from their social media post.

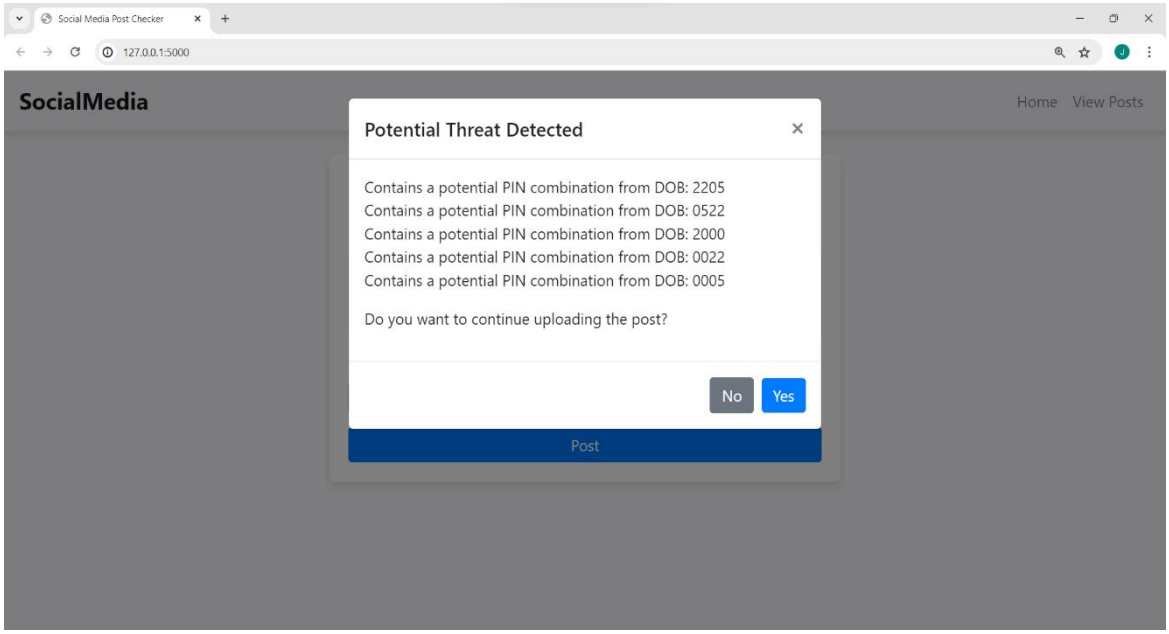

Figure 1: It shows the multiple pin combination from DOB.

This social media post checker identifies a potential privacy threat has shown in figure 1. It found a multiple combination of digits in caption that could be date of birth. The checker then creates multiple variations based on those digits and shows them all as threats. This is because someone could potentially use this information to enter ATM pin to withdraw money or money

transaction. It's important to remember that the accuracy of these warnings depends on how well the application extracts the date of birth, and it might flag unrelated digit combinations too. The severity of these risk also depends on the context of user post. Ultimately, this application helps you identify potential privacy concerns in your social media posts.

Actual Results:

- The Actual result meets the result has expected. However actual results might be less accurate due to limitations in pattern recognition, scope, and over-sensitivity.
- The regex pattern might miss Dobs written in different formats (e.g., 22 May 2000) or misinterpret unrelated digit sequences as DOBs.
- It only analyzes text, missing DOBs hidden within images or mentioned indirectly.
- It assumes all generated PIN combinations are threats, even though real ATM PINs often include a mix of numbers and letters for increased security.

6.2 Experiment / Case Study 2: Detecting Security questions like Dog names

This case study showcases the use of image processing and object detection techniques to identify potential privacy risks related to images posted on social media. Specifically, it focuses on detecting dogs in images, which can inadvertently reveal personal information used in security questions, such as the name of a pet.

Function Explanation

To analyze the image and function (process_image) is implemented to detect an specific objects by using YOLO (You Only Look Once) object detection.

- The function begins by loading the pre-trained YOLO model and its configuration files (yolov3.weights and yolov3.cfg). It also loads the coco.names file, which contains the names of the object classes that the YOLO model can detect.
- The input image is read and converted to a blob format suitable for YOLO processing. This involves resizing the image to the input size expected by the YOLO model (typically 416x416 pixels), normalizing the pixel values, and performing any necessary mean subtraction and scaling.
- The preprocessed image (blob) is fed into the YOLO model, which performs a forward pass to detect objects. The YOLO model outputs several bounding boxes, each with a class label and a confidence score indicating the likelihood that the bounding box contains a particular object.
- The function filters the detected objects to identify only those that correspond to dogs. This is done by comparing the class IDs of the detected objects with the class ID for dogs (as defined in the coco.names file).
- If a dog is detected, the function flags this detection as a potential privacy risk. It prompts the user with a warning, explaining the risk and offering the option to proceed

with or cancel the post. If the user confirms the post despite the warning, the image is displayed with a red border to indicate the potential risk.

Expected Results:

- The function aims to detect dogs within an uploaded image.
- If a dog is detected in the post, it flags this detection as a potential privacy risk, considering that pet names are commonly used as answers to security questions.
- The application will prompt the user with a warning if a dog is detected in the image, explaining the risk and offering the option to proceed with or cancel the post.
- If the user confirms the post despite the warning, the image is displayed with a red border indicating a potential risk.

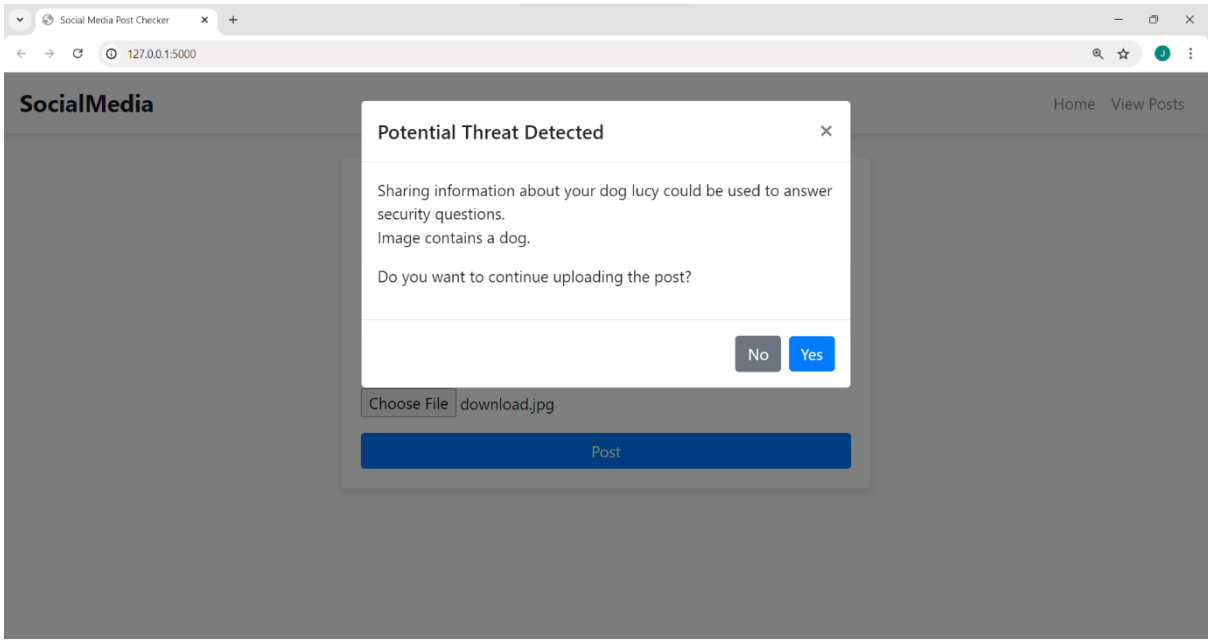

Figure 2: Potential warning about dog detected in the post.

Actual Results

- The actual results demonstrate the capability to detect dogs in images by meeting the expected outcomes.
- However, the accuracy of detection depends on the quality of the image and the effectiveness of the YOLO model.
- The application might detect non-dog objects with similar features which leading to a false positive.
- In some cases, especially with low-light images or images where the dog was partially obscured, the model failed to detect the dog. This highlights the importance of good image quality for accurate detection.

6.3 Experiment / Case Study 3: Detecting Cars and Laptops in user post Images

This case study explores the use of image processing and object detection techniques to identify potential privacy risks associated with images posted on social media. Specifically, it focuses on detecting both cars and laptops in images, which can inadvertently reveal personal information such as cars, and laptops.

Function Explanation

This function work as the same as the case study 2 that is detecting dog in the user post.

Expected result:

- The function aims to detect cars and laptops within an uploaded image.
- If a car or laptop is detected, it flags this detection as a potential privacy risk, considering that cars can reveal personal information such as the make, model, or license plate number, and laptops can display sensitive information.
- The application will prompt the user with a warning if a car or laptop is detected in the image, explaining the risk and offering the option to proceed with or cancel the post.
- If the user confirms the post despite the warning, the image is displayed with a red border indicating a potential risk.

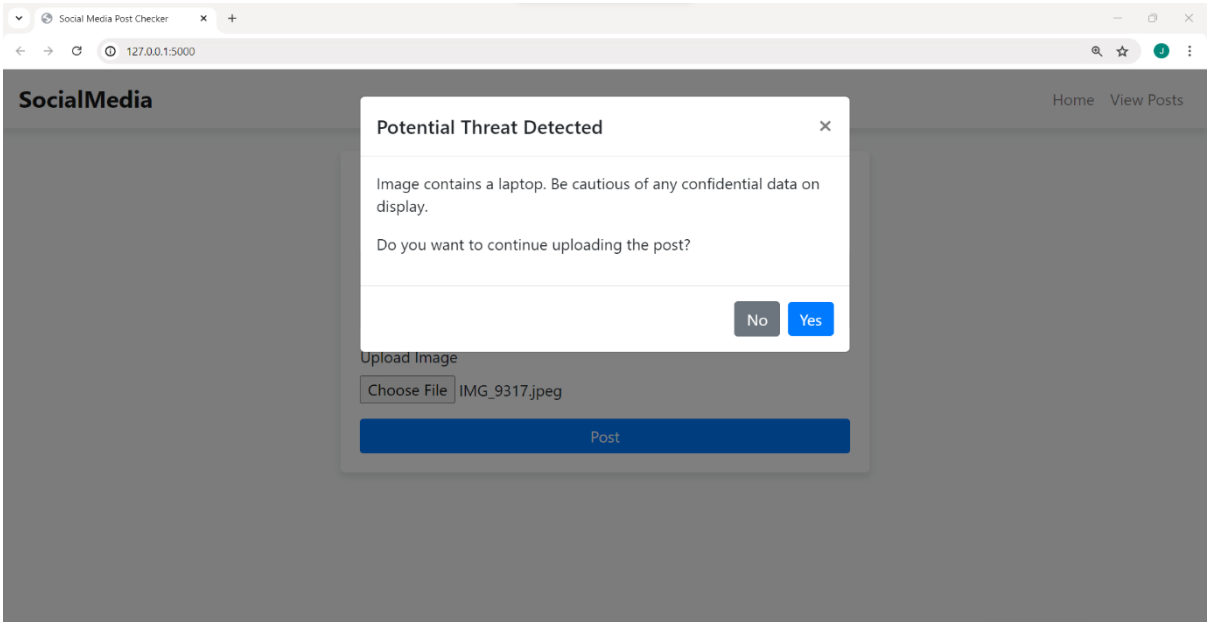

Figure 3: Potential warning about laptop/Computer detected in the post.

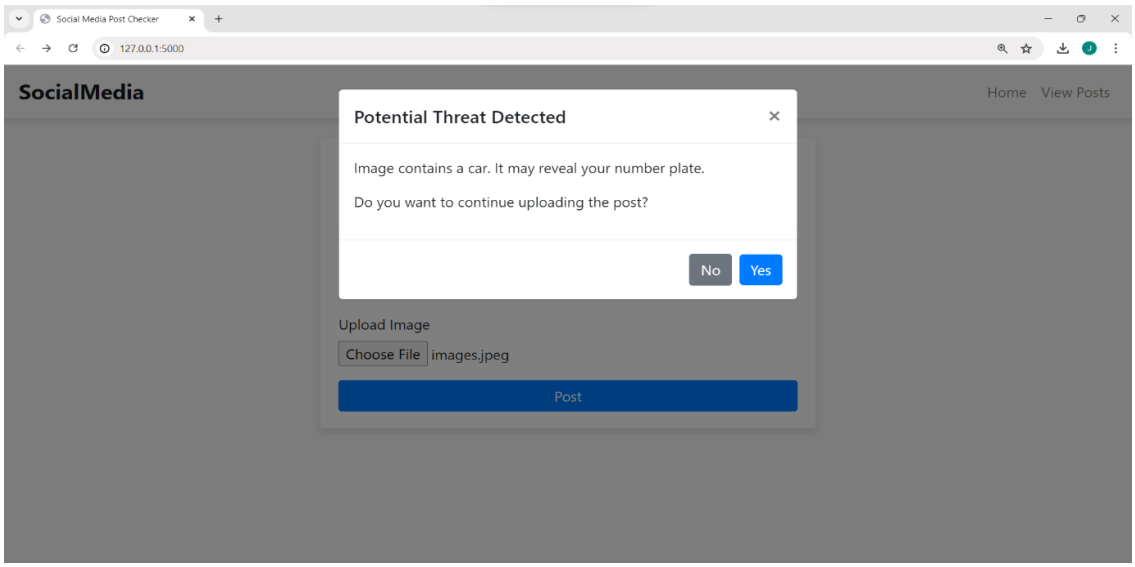

Figure 4: Potential warning about car detected in the post.

Actual Result:

- The function successfully able to detect cars and laptops in various user post on social media.
- The YOLO model was able to accurately detect cars and laptops in images with clear, unobstructed views.
- Some false positives occurred, where objects with similar shapes or features to cars or laptops were incorrectly identified. These instances were minimal and mostly involved images with other vehicles or certain inanimate objects.
- The user interface effectively displayed warnings for detected cars and laptops and allowed users to decide whether to proceed with or cancel the post. The red border around confirmed posts served as a clear visual indicator of potential privacy risks.

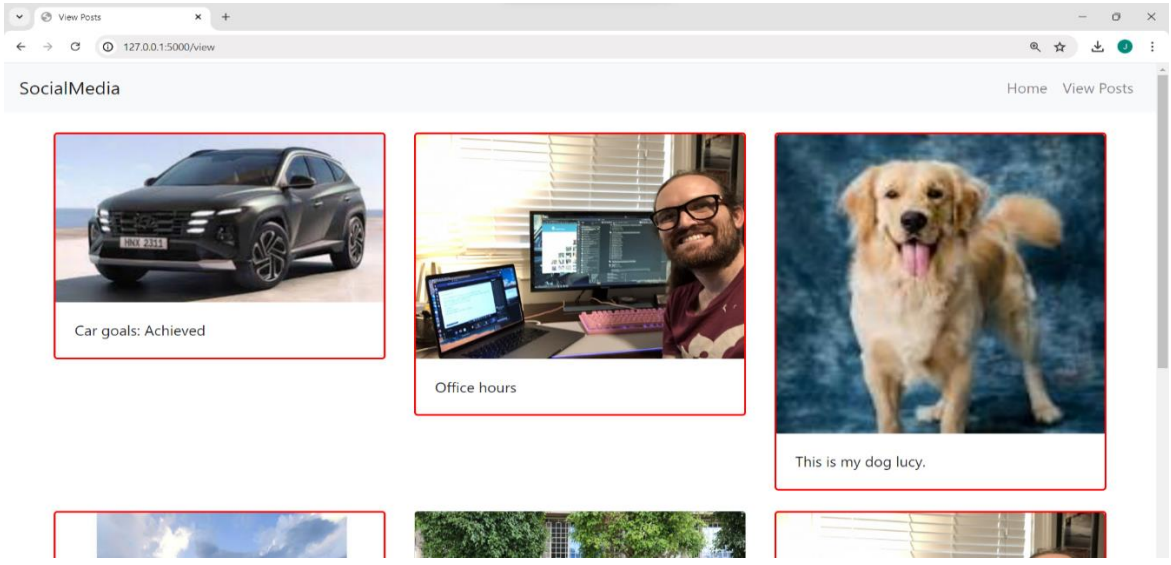

Figure 5: Showing waring in red border that contain threat.

6.4 Experiment / Case Study: Normal Posting for Non-Threatening Images

In this case study, we evaluate the applications behavior when there is no potential privacy threats (cars, laptops, DOB, and security question) are detected in an image or text analysis. The objective is to ensure that images deemed safe for posting are uploaded without any warnings or red borders while posting on social media by providing a seamless user experience for non-threatening content.

Function Explanation

The process_image function identifies potential privacy threats by detecting the specific objects like (cars, laptops, DOB, and security question) in an image. If there is no such objects are found in the application it allows the image to be posted normally. Here, we detail the expected and actual outcomes for non-threatening images.

Expected Result:

- The function aims to detect the cars, laptops, DOB, and security question within an uploaded image and text.
- If no cars, laptops, DOB, and security question are detected, the application will not display any warnings and will post the image as normal.
- The image will be uploaded without any red border, ensuring that non-threatening images are shared seamlessly

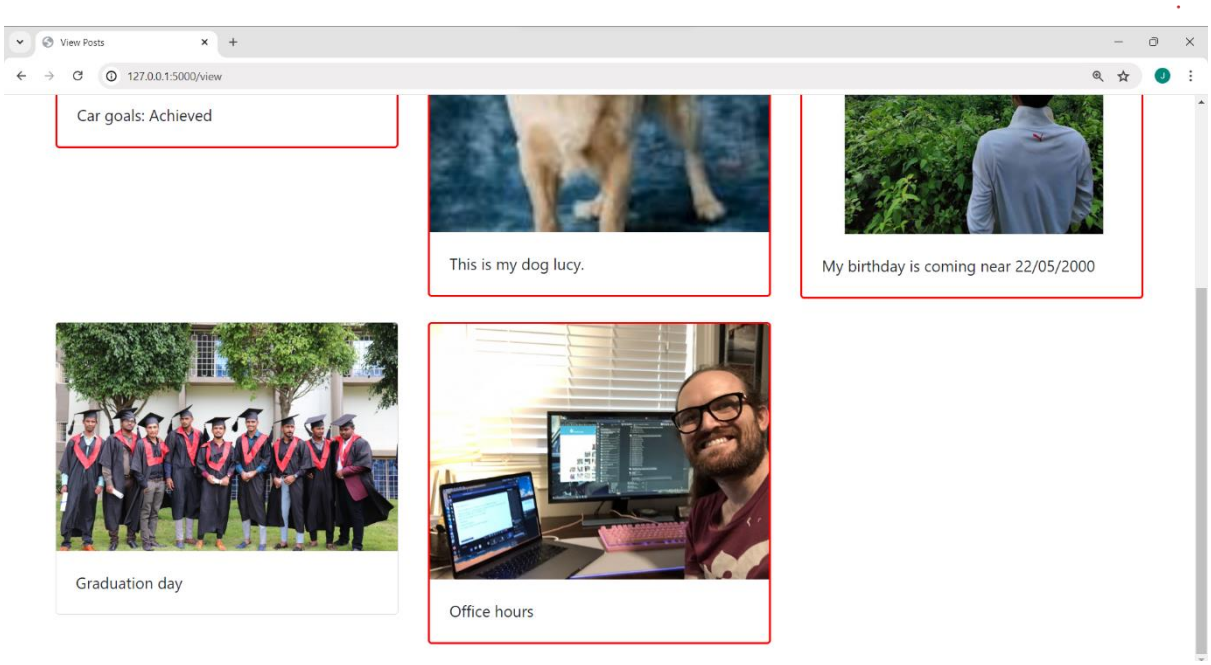

Figure 6: Posting without any waring in red border (Graduation Day post).

Actual Result:

- The function successfully identified images without cars or laptops, meeting the expected outcome of posting these images without any warnings or red borders.
- For images that did not contain any cars or laptops, dog the YOLO model correctly identified them as non-threatening. This application then proceeded to post these images without any interruptions.
- Users experienced a seamless posting process for images that did not contain any privacy threats. This helped in maintaining a smooth user experience for non-sensitive content.
- Images that did not contain cars, dog, laptops and security question like dog name were posted normally without any red borders or warnings by providing a user's with an uninterrupted and smooth sharing experience for safe content.

6.5 Discussion

This section provides a comprehensive analysis of the findings from the conducted experiments and case studies on detecting privacy threats in social media posts. These experiments focused on detecting specific objects (cars, dog and laptops) in images and provided insights into the efficiency and accuracy of the implemented methods. Additionally, the experiments examined the user experience when there are no threats were detected in the post, ensuring a seamless and unobstructed posting process for non-threatening images.

Detailed Analysis of Findings

Experiment / Case Study 1: Predicting ATM PIN Numbers

- Effectiveness: The analysis of the potential ATM PIN numbers from the dates of birth demonstrated the importance of privacy awareness in social media activities. The function effectively identified by various PIN combinations based on extracted dates of birth highlighting the potential risks.
- Limitations: The regex pattern used for date extraction could miss dates written in different formats or misinterpret unrelated digit sequences as dates of birth. Additionally, the assumption that all generated PIN combinations are threats might lead to over-sensitivity.

Experiment / Case Study 2: Detecting Dogs in Images

- Effectiveness: The YOLO-based object detection model successfully identified dogs in images or text with the high accuracy. This is important as sharing information about pets can often be used to answer security questions.
- Limitations: The detection of a dog alone does not always indicate a security risk. The model's performance can be affected by different image resolutions, lighting conditions, and variations in dog breeds.

Experiment / Case Study 3: Detecting Cars in Images

- Effectiveness: The model effectively detected cars in images, alerting users to potential risks such as revealing number plates or other identifying features.
- Limitations: Like dog detection, the context is important. Not every image of a car poses a privacy threat. The model might also miss cars that are partially obstructed or in less common perspectives.

Experiment / Case Study 4: Detecting Laptops in Images

- Effectiveness: The model accurately identified laptops in images, cautioning users about the potential exposure of sensitive information displayed on screens.
- Limitations: The mere presence of a laptop does not necessarily indicate a privacy threat. The model might not recognize all types of laptops or could flag objects that resemble laptops.

Experiment / Case Study 5: Normal Posting for Non-Threatening Images

- Effectiveness: The function accurately identified the non-threatening images and allowed them to be posted without any warnings or red borders in the view page, ensuring a smooth user experience. This case study demonstrated the system's ability to correctly discern between potentially risky and safe content.
- Limitations: While the function performed well in detecting the absence of threats, it relied heavily on the accuracy of the YOLO model. False negatives (undetected threats) could still pose privacy risks.

Critique of Experimental Design

The experimental design was generally effective in achieving the objectives of identifying the potential privacy threats in social media posts. However, there are several areas for improvement and modification were identified:

- The performance of the YOLO model is heavily dependent on the quality and diversity of its training data. The current study did not specify the exact dataset used, which raises concerns about potential biases and limitations in object detection, particularly for objects in diverse environmental conditions.
- The model exhibited some instances of false positives and negatives. Objects that were partially obscured or in low-light conditions were sometimes may not be detected, while sometime irrelevant objects were occasionally detected as a threat. This indicates a need for further tuning and additional training data to enhance the model’s robustness.
- The impact of false positives and negatives on user trust and experience was not adequately explored. Users may become frustrated with false alarms or complacent due to undetected threats, which can undermine the effectiveness of the application.
- The use of regex patterns for text analysis was extremely limiting, despite its simplicity and speed. Regex may fail to capture differences in date formats which further results in missed detections or false positives.

- Regex lacks the ability to understand context, making it unsuitable for detecting more subtle or complex threats. For example, the presence of a keyword like "dog" does always indicate a security risk without additional context.

Modifications and Improvements

- Expanding the training dataset to include a wider range of environmental conditions, object orientations, and scenarios could improve the robustness and generalization of the object detection model.
- Implementing advanced NLP could enhance the system's ability to understand context and detect more nuanced threats.
- Improving the user interface to provide more detailed feedback and practical insights could enhance the user engagement and the effectiveness of the application.

Integration of Findings with Previous Research

Our study integrates the findings from previous research on text and image analysis in social engineering attack detection and extends them to the relay of social media privacy threat detection using object recognition models.

- Extending Beyond Text Analysis: While (Michael Fire, 2014) focused on textual data, our research addresses the gap by implementing an image analysis using YOLO. This approach recognizes the increasing presence of visual content in social media and the potential for privacy threats embedded in images.
- Contextual Awareness: Building on (Jagatic's, 2007) emphasis on context and personalization, our combined detection of dogs, cars, and laptops incorporates the importance of contextual analysis in determining the relevance of detected objects as potential privacy threats.
- Comprehensive Frameworks: Similar to (Zeadally's, 2015) comprehensive framework for identity the deception, our study proposes a multifaceted approach to threat detection by combining the object recognition with an contextual and user behavior analysis to enhance the accuracy and reliability of the system.
- Advanced Image Analysis: Following (Bo Mei's, 2018) use of CNNs for inference attacks, our application of YOLO leverages the strengths of neural networks for real-time object detection, addressing the need for efficient and accurate analysis of visual data.

7 Conclusion and Future Work

Research Question and Objectives

Our research aimed to answer the question: "Can integrating the YOLO model with natural language processing for content analysis provide an effective early warning system to safeguard social media platform users against social engineering attacks?" The primary objectives were to develop an image processing system capable of identifying specific threats (dogs, cars, and laptops) in user uploaded images which provide an appropriate warning and maintain normal functionality for non-threat images.

To achieve these objectives, a system was developed using the YOLO (You Only Look Once) object detection model to identify potential security threats in images. The system was designed to detect dogs, cars, and laptops due to their potential to expose personal information. We have implemented a web application that allowed users to upload images and captions like social media which were then analysed for threats. The system provided warnings for identified threats and marked the posts visually with the red border. Non-threat images were uploaded without any alterations.

The application demonstrated effective detection of common objects and potential security threats in user-uploaded images, as well as the ability to analyze text for personal identifiers that could be exploited in social engineering attacks. The combination of image and text analysis significantly improves the prediction and prevention capabilities of the system, offering a comprehensive approach to cybersecurity.

This research contributes to the field of cybersecurity by providing a practical tool for predicting social engineering attacks, emphasizing the importance of integrating multiple data analysis techniques. The system's real-time detection capability represents a significant advancement in the proactive defence against social engineering threats, making it an asset for both individuals and organizations.

Key Findings

- The system demonstrated a high precision in detecting the targeted objects (dogs, cars, and laptops) in user-uploaded images on social media.
- Appropriate warnings were generated for identified threats, enhancing user awareness of potential privacy risks.
- The system correctly identified non-threat images, uploading them without any unnecessary warnings or visual alterations.
- The implementation of visual markers (red borders) effectively indicated posts containing potential threats, aiding user decision-making.

Future Work

There are various areas where further research and development could considerably improve the capabilities and efficacy of the threat detection system developed during this work.

Enhanced Date of Birth Format Detection

The Future work should also include the ability to recognize a wider variety of date of birth (DOB) formats. This enhancement would improve the system ability to detect sensitive information embedded in text more accurately. Recognizing formats such as 22 May 2000, or May 22, 2000, can help in identifying potential risks more comprehensively. This improvement is particularly relevant as users often share their DOB in various formats on social media.

Advancing in Image Processing for Privacy Risk Detection:

Improvements in image processing will need the use of more modern versions of object detection, which provide greater accuracy and speed. Image enhancing approaches will improve identification resilience in low-light or obstructed photos, and context-aware models will be created to reduce false positives by considering objects surrounds and placements. The list of detectable objects will be expanded to include items that may constitute a privacy concern, such as house keys, credit cards, or personal documents, and models will be trained to detect sensitive text within photos, such as names or addresses on letters or documents.

User Experience and Interface Enhancements:

To improve the user experience, a more advanced warning system will be developed to measure the severity of detected threats and customize alerts appropriately, while providing educational prompts will provide users with privacy-enhancing advice. allowing users to submit false positives or missed threats which will help to improve the accuracy of the detection.

8 References

Bo Mei, Y. X. R. L. H. L. X. C. &. Y. S., 2018. Image and Attribute Based Convolutional Neural Network Inference Attacks in Social Networks. *IEEE Transactions on Network Science and Engineering,* 7(2), pp. 869-879.

Dhruvi D. Gosai, H. J. G. &. P. H. S. J., 2018. A Review on a Emotion Detection and Recognization from Text Using. *nternational journal of applied engineering Research,* Volume 13, pp. 6745-6750.

Farhadi, J. R. $. A., 2018. YOLOv3: An Incremental Improvement. *arXiv (Cornell University).*

Iamnitchi, I. K. &. A., 2017. Privacy and security in online social networks: A survey. *Online social networks and media,* pp. 1-21.

Index, P. P., n.d. [Online] Available at: https://pypi.org/project/Flask/ [Accessed 3 May 2024].

Index, P. P., n.d. *numpy.* [Online] Available at: https://pypi.org/project/numpy/ [Accessed 4 May 2024].

Index, P. P., n.d. *opencv-python.* [Online] Available at: https://pypi.org/project/opencv-python/ [Accessed 3 May 2024].

JetBrains, n.d. *The Python IDE for data science and web development.* [Online] Available at: https://www.jetbrains.com/pycharm/download/?section=windows

Joseph Redmon, S. D. R. G. &. A. F., 2016. You Only Look Once: Unified, Real-Time Object Detection.

Kaabouch, F. S. &. N., 2019. Social Engineering Attacks: A Survey. *Future Internet,* 11(4), p. 89.

Kaggle, n.d. *coco-names.* [Online] Available at: https://www.kaggle.com/code/malikachhibber/coco-names [Accessed 10 May 2024].

Kaggle, n.d. *Yolov3 Weights.* [Online] Available at: https://www.kaggle.com/datasets/shivam316/yolov3-weights [Accessed 10 May 2024].

Kaggle, n.d. *yolov3.cfg.* [Online] Available at: https://www.kaggle.com/datasets/julienelk/yolov3cfg/data [Accessed 10 May 2024].

Katharina Krombholz, H. H. M. H. &. E. W., 2015. Advanced social engineering attacks. *Journal of Information Security and Applications,* pp. 113-122.

Magazine, C., 2020. *Cybercrime To Cost The World $10.5 Trillion Annually By 2025.* [Online] Available at: https://cybersecurityventures.com/cybercrime-damages-6-trillion-by-2021/#:~:text=If%20it%20were%20measured%20as,after%20the%20U.S.%20and%20China. [Accessed 28 June 2024].

Michael Fire, R. G. &. Y. E., 2014. Online Social Networks: Threats and Solutions. *IEEE Journals & Magazine | IEEE Xplore.*

Michael Fire, R. G. &. Y. E., 2014. Online Social Networks: Threats and Solutions. *IEEE Communications Surveys & Tutorials,* 16(4), pp. 2019-2036.

Nelson Duarte, N. C. &. T., n.d. Social Engineering: The art of attacks.

Sennovate, 2023. *Understand How Social Media Is Fuelling Social Engineering Attacks.* [Online]
Available at: https://sennovate.com/understand-how-social-media-is-fuelling-social-engineering-attacks/#:~:text=In%20this%20type%20of%20social,is%20known%20as%20contact%20spamming.
[Accessed 13 June 2024].

Tom N. Jagatic, N. A. J. M. J. &. F. M., 2007. Social phishing. *Communications of the ACM,* 50(10), pp. 94-100.

Wei Fang, L. W. &. P. R., 2019. Tinier-YOLO: A Real-Time Object Detection Method for Constrained Environments. *IEEE Access,* Volume 8, pp. 1935-1944.

Zeadally, M. T. &. S., 2015. Detecting and Preventing Online Identity Deception in Social Networking Services. *IEEE Internet Computing,* 19(3), pp. 41-49.