**HỌC VIỆN KỸ THUẬT MẬT MÃ**

**KHOA CÔNG NGHỆ THÔNG TIN**



**ĐỒ ÁN TỐT NGHIỆP**

Đề tài:

### Phát hiện mã độc trên Android dựa trên biểu diễn đặc trưng đồ thị

*Giảng viên hướng dẫn*: **TS. LÊ ĐỨC THUẬN**

*Sinh viên thực hiện*: Nguyễn Văn Thanh

*Mã sinh viên:* CT050446

*Khoa:* Công nghệ thông tin

*Lớp:* CT5D

*Khóa:* 2020-2025

**Hà Nội, tháng 12 năm 2024**

**ĐỀ CƯƠNG**

### **Chương 1: Giới thiệu tổng quan**

#### **1.1. Lý do chọn đề tài**

* Nêu vấn đề: Sự gia tăng các cuộc tấn công mã độc trên hệ điều hành Android và mức độ nguy hiểm đối với người dùng.
* Phân tích tầm quan trọng của việc sử dụng đặc trưng đồ thị để biểu diễn và phân tích mã độc.
* Đưa ra lý do lựa chọn phương pháp Graph Neural Network (GNN) làm nền tảng phát hiện mã độc.

#### **1.2. Mục tiêu nghiên cứu**

* Trình bày mục tiêu chính của đồ án, bao gồm:
  + Xây dựng một mô hình GNN hiệu quả để phát hiện mã độc Android.
  + Tích hợp mô hình vào một ứng dụng cho phép phân tích mã độc từ các file APK tải lên.

#### **1.3. Phạm vi nghiên cứu**

* Giới hạn nghiên cứu:
  + Phân tích mã độc tĩnh.
  + Phát hiện mã độc trên Android thông qua file APK.
  + Sử dụng đặc trưng đồ thị của mã nguồn và quyền truy cập của ứng dụng để huấn luyện mô hình.

#### **1.4. Tổng quan các nghiên cứu liên quan**

* Tóm tắt các phương pháp truyền thống trong phát hiện mã độc (dựa trên chữ ký, phân tích hành vi).
* Trình bày về tiềm năng và ưu điểm của Graph Neural Network trong xử lý dữ liệu đồ thị phức tạp.

### **Chương 2: Phương pháp nghiên cứu và triển khai**

#### **2.1. Thu thập và xử lý dữ liệu**

* Thu thập bộ dữ liệu:
  + Tìm kiếm và tải về các file APK, bao gồm cả ứng dụng mã độc và không mã độc.
* Xử lý dữ liệu:
  + Sử dụng APKTool trích xuất đặc trưng đồ thị từ mã nguồn (bao gồm quyền truy cập, API call, và luồng dữ liệu).
  + Dùng thư viện PyTorch geometric để chhuyển đổi dữ liệu thành định dạng phù hợp để huấn luyện mô hình GNN.

### **2.2. Tìm hiểu các mô hình Graph Neural Network (GNN)**

#### **Tìm hiểu lý thuyết và các loại GNN dự kiến**

#### **Graph Convolutional Network (GCN)**:

* + Là một trong những mô hình GNN cơ bản, áp dụng phép tích chập (convolution) lên đồ thị để tổng hợp thông tin từ các nút lân cận.
  + GCN phù hợp để xử lý các đồ thị nhỏ hoặc đơn giản.
* **Graph Attention Network (GAT)**:
  + Cải tiến từ GCN, sử dụng cơ chế **attention** để gán trọng số khác nhau cho các nút lân cận dựa trên mức độ quan trọng của chúng.
  + GAT rất phù hợp với các đồ thị phức tạp, chẳng hạn như đồ thị trích xuất từ mã nguồn của ứng dụng.

#### **Mục tiêu**:

* Hiểu rõ ưu, nhược điểm của từng mô hình.
* Lựa chọn mô hình phù hợp nhất cho bài toán phát hiện mã độc.

### **2.3. Xây dựng và huấn luyện mô hình phát hiện mã độc**

#### **Thiết kế kiến trúc mô hình**

* **Phân tích đồ thị đặc trưng**:
  + Sử dụng các mô hình GNN để học từ các đồ thị biểu diễn đặc trưng của ứng dụng (API call, quyền truy cập, luồng dữ liệu).
* **Lựa chọn siêu tham số (hyperparameters)**:
  + Điều chỉnh các thông số như:
    - **Learning rate**: Tốc độ học của mô hình.
    - **Số layer**: Số tầng trong mạng GNN.
    - **Batch size**: Kích thước dữ liệu trong mỗi lần huấn luyện.

#### **Huấn luyện mô hình**

* **Chia dữ liệu**:
  + Tập dữ liệu được chia thành 3 phần:
    - **Train**: Dùng để huấn luyện mô hình.
    - **Validation**: Dùng để kiểm tra trong quá trình huấn luyện, giúp điều chỉnh siêu tham số.
    - **Test**: Dùng để đánh giá cuối cùng hiệu quả của mô hình.
* **Đánh giá mô hình**:
  + Sử dụng các chỉ số phổ biến như:
    - **Accuracy**: Tỷ lệ dự đoán đúng.
    - **Precision**: Tỷ lệ dự đoán đúng trong các dự đoán là mã độc.
    - **Recall**: Tỷ lệ phát hiện đúng mã độc trong tổng số mã độc thực tế.
    - **F1-score**: Sự cân bằng giữa Precision và Recall.

### **2.4. Phát triển ứng dụng phát hiện mã độc**

#### **Xây dựng ứng dụng**

* Tạo một ứng dụng **web** tích hợp mô hình GNN.
* Tích hợp mô hình GNN vào hệ thống .
* Xây dựng giao diện người dùng bằng React.js

#### **Chức năng chính**

1. **Tải file APK**:
   * Cho phép người dùng tải lên các file APK để phân tích.
2. **Phân tích mã độc**:
   * Ứng dụng sử dụng mô hình GNN để phát hiện mã độc trong file APK.
3. **Báo cáo chi tiết**:
   * Hiển thị các thông tin:
     + **Mức độ nguy hiểm**: Đánh giá mức độ rủi ro của ứng dụng.
     + **Hành vi đáng ngờ**: Các hoạt động bất thường được phát hiện.
     + **Quyền truy cập**: Liệt kê các quyền truy cập có khả năng bị lạm dụng.

### **Chương 3: Kết quả và Đánh giá**

#### **3.1. Kết quả huấn luyện mô hình**

* Trình bày kết quả đạt được sau khi huấn luyện mô hình GNN, bao gồm:
  + Độ chính xác, độ nhạy, và các chỉ số đánh giá khác.
  + So sánh hiệu quả giữa các mô hình GNN được thử nghiệm.

#### **3.2. Đánh giá hệ thống**

* So sánh với các phương pháp
* Nêu rõ các hạn chế của mô hình và hệ thống.

#### **3.3. Đề xuất cải tiến**

#### **3.4. Kết luận**

* Tóm tắt những kết quả chính đạt được.
* Đánh giá khả năng ứng dụng thực tế của mô hình và hệ thống.

1.2 Phát hiện mã độc trên android

1.3 Các bộ dữ liệu được dùng trong nghiên cứu

1.4 Phương pháp đề xuất

Chương 2 Tìm hiều mô hình học sâu kết hợp

1. Tìm hiều mô hình GCN

2. Tìm hiều mô hình GAT

Chương 3: thực nghiệm

3.1 Chuẩn hóa đặc trưng

3.2 Các tham số trong mô hình

3.3 Kịch bản thử nghiệm  
 - tạo các bộ dữ liệu chuyển vào CNN

3.4 Kết quả

3.5 Đánh gias