

マルウェア動的解析で要求される対話的イベントの抽出と LLMによる自動応答可能性の検討

塙 剛生^{1,a)} 山岸 伶^{2,3} 藤井 翔太¹

概要：マルウェアには、ダイアログのボタンクリックなどの対話的な操作の要求（以下、イベントと呼ぶ）に応答することによって動作が進行するものが存在する。このため、マルウェア動的解析を自動化するには、各イベントに対して適切な操作を自動で実行する必要がある。他方で、そうしたイベントは多種多様であり、ルールに基づく現行のマルウェア動的解析の自動化システムでは、網羅的に対応することが現実的ではない。そこで本研究では、動的解析の完全自動化を目指し、柔軟な操作を実現する手法として、LLM（大規模言語モデル）の活用可能性を検討した。まず、体系的に整理されていないイベントの種類とイベントに対する適切な操作を明らかにするため、154件の解析レポートをコーディングし、508件のイベントとイベントに対する解析操作を抽出した上で、イベントの分類とデータセットの構築を行った。次に、データセットを基に、LLMを用いてイベントから解析操作を生成し、コーディングで抽出した操作との一致度評価を実施した。この結果、LLMが68.19%の精度で解析操作を生成できることが実証され、自動応答への活用可能性を示した。さらに、LLMを動的解析に活用する際の課題と今後の研究可能性を導出した。

Extraction of Interactive Events in Dynamic Malware Analysis and Evaluation of LLM-Based Automated Response Capabilities

GOKI HANAWA^{1,a)} REI YAMAGISHI^{2,3} SHOTA FUJII¹

Abstract: Some malware activates behavior by responding to interactive requests such as dialog button clicks. Therefore, automating dynamic malware analysis requires automatic execution of accurate actions for each event. However, traditional rule-based systems struggle with malware that demands actions beyond predefined rules. To address this limitation, we explored the potential of large language models (LLMs) to flexibly emulate analyst actions in dynamic malware analysis. In this study, we conducted a coding of 154 analysis reports, extracted 508 events requiring actions, and classified them to build a dataset linking events to analysis actions. We prompted LLMs to generate appropriate analysis actions and evaluated its accuracy against the coded analysis actions using this dataset. As a result, we demonstrated LLMs emulated analysis actions with 68.19% accuracy, and the capability of LLMs. Furthermore, we summarized the issues when using LLM as a automated dynamic malware analysis, and recommendations for future research possibilities.

1. はじめに

マルウェア動的解析は、解析環境においてマルウェア検体を実際に動作させることによって、その挙動を明らかにする手法である。ここで、実際の組織には日々マルウェア検体が着弾することから、限られた人員がすべての検体を

手作業で解析するのは現実的ではない。このため、解析者による手動解析に加え、サンドボックスなどの自動動的解析を導入し、基本的な解析を自動実行することで全体の効率化が図られている[1]。

ここで、マルウェアの中には、ダイアログのボタンクリックやClickfix[2]でのキー入力などの操作要求（以下、イベントと呼ぶ）に適切に応答した場合にのみ、動作が進行して悪性行動に至るものも存在する。このようなイベントを伴う挙動は多種多様であるのに対して、従来の自動解析

¹ 株式会社日立製作所 Hitachi, Ltd.

² 国立研究開発法人 情報通信研究機構 NICT

³ 早稲田大学 Waseda University

a) goki.hanawa.rc@hitachi.com

技術の多くは単純なキーワードマッチングに依存したルールベースの手法にとどまっている。このため、従来手法は定義外のイベントに対応できず、完全自動化の実現には限界がある。今後は、より広範なイベントに対して柔軟かつ適切な操作を実行できる自動解析技術が求められる。このためには、解析環境上に表示されるイベントを正確に認識し、解析に必要な操作を判断する技術が必要である。近年、画像と言語の統合的な理解能力を有する大規模言語モデル (Large Language Model: LLM) が注目されており、これを用いることでイベントの認識や必要操作の判断を実現できる可能性がある。本研究が見据える最終的な目標はマルウェア解析の完全自動化であり、その第一歩として本稿では LLM による柔軟な応答生成への活用可能性を検証する。

LLM の活用可能性を検証するにあたっては、動的解析中に表示されるイベントから、LLM がどの程度適切な操作を抽出できるかを評価する必要がある。しかし、我々の知る限り、イベントや解析操作を体系的に整理した先行事例は存在しない。そこで本稿では、動的解析中に観測されるイベントおよび解析操作の実態を明らかにし体系的に整理したうえで、LLM の検証を実施する。これらを踏まえ 2 つの研究課題（以降、RQ）を設定する。

RQ1. 動的解析中にどのようなイベントが表示されるか

RQ2. LLM はイベントから解析に適した操作を抽出可能か

これら RQs の解決のために、まず、動的解析レポート中に含まれるイベントを有する画面（以下、イベント画面と呼ぶ）を対象としたコーディングを実施することで、イベントおよび解析操作の抽出・整理を行う。次に、イベント画面と解析操作を紐づけたデータセットを作成したうえで、データセットを基に LLM がイベントを適切に認識し、適切な解析操作を抽出できるか評価することで、LLM の活用可能性の検証を実施する。

本稿における主要な貢献は以下の通りである。

- 154 件のレポートに含まれるイベント画面を対象としたコーディングにより、延べ 508 件のイベントと解析操作を抽出した。加えて、重複を除く 202 件のイベントと解析操作を基にデータセットを作成した。
- データセットを基に、LLM を用いてイベント画面から解析操作を生成し、コーディングで抽出した操作との一致度評価を実施した。その結果、LLM は 68.19% の一致度で解析操作を生成できることが確認された。また、ルールベースな従来手法では対応できない Clickfix 等の複雑なイベントに LLM は対応可能であり、自動応答における活用可能性が示唆された。
- LLM による自動応答の課題として、イベントから解析操作の抽出に必要な情報の認識不足等が特定された。またこれら課題に対して、プロンプト設計の工夫を中心とした今後の研究可能性を導出した。

2. 研究背景

2.1 マルウェアの動的解析における自動化

マルウェアの動的解析は、検体を実際に動作させ、その挙動を観察することでマルウェアの性質や機能を明らかにする手法である。解析の実施方法には、解析者が操作して逐次確認する手動解析と、専用の環境で挙動を自動的に記録する自動解析が存在する。いずれの場合も、安全に実施するためには外部環境への影響を防ぐ隔離環境が不可欠であり、一般にサンドボックスが利用されている。サンドボックスにはローカル型 (Cuckoo Sandbox や DRAKVUF [3] 等) やオンライン型 (ANY.RUN [4], JoeSandbox [5] 等) があり、一部にはイベントに対して解析操作を実行する自動化機能が備わっている。例えば、Cuckoo Sandbox では、事前に作成されたキーワード^{*1}に基づくダイアログ内のボタンの自動クリック機能が導入されている。DRAKVUF も、キーワード^{*2}ベースでの自動クリック機能が実装されている [6]。また、Joe Sandbox では、フィッシングサイトの解析時に画面情報からボタンやリンクの位置・形状を認識することによる自動クリックが実現されている [7]。

ただし、これらの既存手法はクリック操作の自動化に留まっており、マウスによる対話的操作を通じたファイル実行や情報入力など多様な操作には十分対応できていない。このため、より広範かつ複雑な操作を要するイベントへの適応が課題である。本研究では、LLM を活用した柔軟な応答生成の有効性を検証することにより、従来手法では困難であったイベント対応の限界を突破し、動的解析自動化のカバレッジ拡大を目指す。

2.2 LLM を活用したマルウェア解析と自動化に関する研究

LLM に関する研究は、サイバーセキュリティ分野においても急速に進展している。マルウェア解析への LLM の活用はすでに検討されており、Fujii ら [8] はマルウェアの静的解析のために、Wong ら [9] は、回避型マルウェアへの対策のために LLM の活用を検証している。動的解析でのイベント対応自動化は画面に表示されるイベントの認識および判断を要し、主に言語情報を対象とする既存研究とは要件が異なる。したがって、LLM 活用については別途検証する必要があり、本研究ではその検証を行う。

LLM を活用した自動化技術はサイバーセキュリティ分野に限らず、近年高度なタスクへの応用が進んでいる。Zhang ら [10] および Wen ら [11] は、Android 端末のアプ

^{*1} <https://github.com/cuckoosandbox/cuckoo/blob/master/cuckoo/data/analyser/windows/modules/auxiliary/human.py>

^{*2} https://github.com/tklenyel/drakovuf/blob/main/src/plugins/hidsim/gui/vmi_win_gui_filter_definitions.h

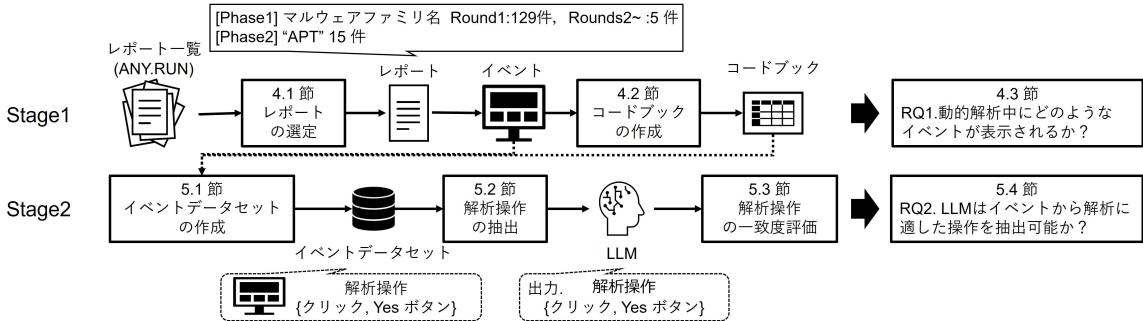


図 1 本研究の全体像

リ操作を LLM で自動実行する手法を提案している。また, Gao ら [12] は, Android 環境より複雑かつ自由度の高い Windows アプリに対する LLM を用いた自動操作を検討している。LLM を活用した自動化は他分野で成果を示しているが, 例えは, ClickFix のような複雑な操作を要するイベントなど, 動的解析中に表示されるイベントは多様で予測困難である。そのため, 先行研究を踏まえたうえで, まず動的解析中に表示されるイベントを把握し, LLM の活用可能性をイベント別に検証する必要がある。

3. 研究方針

3.1 研究設計

本研究では, RQs の解明を目的に二段階の実験を行う。研究の全体像を 図 1 に示す。まず, 第一段階 (Stage1) として, RQ1への回答のためにイベントと解析操作を抽出・整理する。その後, 第二段階 (Stage2) として, Stage1で抽出したイベントと解析操作を紐づけたデータセットを作成し, データセットを基に LLM の自動応答への活用可能性を検証する。

4. Stage1 : コーディング

Stage1 ではコーディングにより, 動的解析中のイベント・解析操作を体系的に整理した。コーディングは, データを細かい単位に分割し, それぞれにラベル (コード) を付与することで, データ内の傾向や特徴を抽出する方法である。コーディング対象は, ANY.RUN [4] の解析レポートに含まれるイベント画面とした。イベント画面の取得方法として, 自前の環境で解析処理を録画する方法と, 公開されているオンラインサンドボックスの解析レポートから収集する方法が考えられる。前者は動画で詳細に記録できる利点があるが, 解析可能な検体数が限られ, コーディングの範囲が狭くなるため, 本研究では多様なイベント画面を広範に収集できる後者を採用した。オンラインサンドボックスとして, 調査対象となるレポートの選定に活用できるトレンドレポート [13] を公開している ANY.RUN を選定した。

本研究では, 後述する検索条件に基づき, コーディング

対象となる解析レポートを選定した (レポートの選定)。次に, 解析レポートに含まれるイベント画面を対象に著者ら 2 名がそれぞれ独立にコーディングを行い, コードの内容について議論した上で, コードブックを作成した (コードブックの作成)。レポート選定およびコードブック作成は 2 段階の Phase を通じて行った。Phase1 ではトレンドレポートのマルウェアファミリに基づき解析レポートを選定したが, ばらまき型のマルウェアに偏る傾向があった。そこで Phase2 では, このバイアスを緩和するため標的型攻撃に用いられるマルウェアの解析レポートを選定した。また, Phase1 では, レポートの選定とコードブック作成を 1 ラウンドとするラウンド制を採用し, 理論的飽和 [14] に達する (新規コードが特定されなくなる) まで反復した。この過程により, 著者間での綿密な議論と十分な情報抽出を実現した。以降の各項では, Stage1 の各手順を詳述した後に, コーディング結果を示す。

4.1 レポートの選定

Phase1 では, ANY.RUN のトレンドレポートに掲載されるマルウェアファミリのうち, 著者らが定義した検索条件を満たす解析レポートが存在した 129 ファミリを対象とした。Phase1 におけるラウンド 1 では, 広範なデータに基づく基本的なコーディング枠組みの構築を目的とし, 129 ファミリのファミリ毎に 1 件ずつ, 計 129 件のレポートを選定した。ラウンド 2 以降では, 理論的飽和の確認を目的とし, 129 ファミリから無作為に選ばれた 5 ファミリについて, それぞれ 1 件ずつ, 計 5 件のレポートを追加で選定した。Phase2 では, ファミリに基づく選定では標的型攻撃で用いられるマルウェアが含まれない可能性を懸念し, さらに網羅性を高めるために別途 “APT” という検索ワードを設定し, ヒットした計 15 件のレポートを選定した。

検索条件

- Verdict: Malicious
- Tag: コーディング対象内のファミリ名 (Phase1), “APT”(Phase2)
- Date To: 2024/12/1 から 2025/5/31 の期間内でランダムに選択された日程

表 1 コードブックの一部

大カテゴリ	中カテゴリ（イベントタイプ）	カテゴリ	コード数	代表的なコード
イベント	解析用アプリ	解析操作の目的	1	ファイル情報の確認
	（システム監視やリソース状況の把握など、解析や管理を目的としたアプリケーション）	内容・役割	1	ファイル情報の表示
	アプロ（特定の分類に属さない、一般的なユーザー向けソフトウェア）	解析操作の目的	4	ファイルを開く、テキストのコピー、ログイン処理の実行、ファイルのダウンロード、ライセンスの同意
	ブラウザメニュー（Web ブラウザ上で操作時に表示されるポップアップや設定メニュー）	内容・役割	5	ホームページの表示、ファイルの選択、ログインの要求、ドキュメントの表示、ライセンスの確認
	ブラウザページ（ブラウザで表示される Web ページのコンテンツ画面）	解析操作の目的	2	ファイルのダウンロードの許可、ダウンロードの完了
	コマンドライン（文字ベースでコマンドを入力・実行するインターフェース）	内容・役割	2	ダウソードの許可、ダウンロードの完了
	コンテキストメニュー（ファイルやデスクトップなどを右クリックした際に表示されるメニュー）	解析操作の目的	7	ファイルのダウンロード、CAPTCHA 認証の通過、ClickFix の操作実行、Cookie の承認、ページ全体の確認
	デスクトップ（OS 起動後に最初に表示されるユーザーの作業領域画面）	内容・役割	6	ダウソードサイト、認証情報の入力、CAPTCHA 認証、Cookie の設定、予約サイト、フォルダ内の一覧表示
	ダイアログ（システムやアプリケーションがユーザーへ通知や選択を求めるポップアップ画面）	解析操作の目的	4	コマンドラインを開く、一時停止の解除、ファイルの実行、攻撃コマンドの実行
	ファイルマネージャー（ファイルの閲覧・操作・解凍などを行うためのアプリケーション）	内容・役割	4	エラー文、RAT のコマンド選択、管理者権限で実行、ファイルの実行、ファイル名の変更、テキストのベース
	インストーラー（ソフトウェアを導入する過程で表示されるセットアップ用画面）	解析操作の目的	1	ファイル操作内容の表示
	マクロ付きファイル（マクロが組み込まれており、実行時に警告や操作が発生するファイル）	内容・役割	4	ファイルの実行、ファイル名の変更、フルダ解凍、ファイル・フォルダを開く
	メール（メールの送受信や閲覧を行うアプリケーション）	解析操作の目的	3	ホームページ上のファイルの表示、リネーム処理、ファイル開封の指示
	マルウェア関連アプリ（マルウェア自体の設定・ビルト・制御を行うための専用アプリケーション）	内容・役割	12	ダイアログを開じる、ファイルの置換、ファイル操作の終了、ファイルを開く、ファイル名の変更
ファイル名を指定して実行（Windows の “Win+R” で呼び出され、プログラムやバスを入力して実行する GUI）	解析操作の目的	10	エラー文、警告、設定情報の選択、情報の入力、ファイル操作の選択、UAC、脅迫文、不明（文字化け）	
解析操作	クリック	実行箇所	2	ファイルを開く、実行、ファイル名の変更
	ダブルクリック		2	フォルダ内の表示、リネーム処理
	右クリック		3	インストール、アンドィベート、ボリサーの承認
	キー入力	キー	4	インストールに係る説明、進捗の表示、設定の確認、ボリサーの表示
	ホットキー	キー	5	マクロの有効化・実行、ファイル全体の確認、ファイルの保存、ファイルのダウソード、警告を閉じる
	スクロール	方向	8	ファイルの開封で示唆、エラー文の表示、データ損失の警告、アプリ更新の促進、ダウソードリンク誘導
			13	添付ファイルの開封、実行

4.2 コードブックの作成

本研究では、選定された解析レポートに基づき、著者らがそれぞれ独立にイベント画面を対象にコーディングし、その妥当性をレビューした上で、コードブックを作成した。コーディングでは、第1著者と第2著者が独立にイベント画面を確認し、それぞれの主観的視点に基づいてラベルを付与した。なお、第1著者と第2著者はそれぞれサイバーセキュリティ研究歴1年と6年、コーディング経験1回と6回であった。

レビューでは、著者ら2名がそれぞれ作成したコードブックを持ち寄り、コード表現や基準のすり合わせを目的とした議論を行い、コードブックを統合した。コーディングおよびレビューは、Phase1, Phase2の順に実施され、Phase2の終了時点でコードブックを完成させた。

4.3 RQ1：結果

表1に作成したコードブックの一部を示す。Phase1は新たなコードが発見されなくなったラウンド3で終了とし、結果として154件のレポートをコーディングし、延べ508件のイベントと、そのイベントに対する解析操作が抽出された。コードブックでは、“イベント”、“解析操作”的2つに大別し、抽出されたコードを階層的に分類した。なお、表1では、スペースの都合上、各カテゴリの代表的なコードのみ記載する。

大カテゴリ：イベント

大カテゴリ“イベント”では、中カテゴリにイベントの種類、カテゴリに、その解析操作を選択した際の目的を表す“解析操作の目的”とイベントの解釈や認識の内容を表す“内容・役割”とし、コードを抽出した。大カテゴリ“イベント”では、14種類の中カテゴリ（イベントタイプ）をそれぞれ定義した上で抽出した。

中カテゴリ“ダイアログ”や“インストーラー”では、“ダイアログを開じる”や“インストールの進行”といった、ユーザーの明示的な意思決定や確認操作が求められることが確認された。これらのイベントタイプは、マルウェアが対話的な操作を必要とする典型的な例であり、自動化のためにはイベントの内容に応じた適切な操作選択が重要と推察される。

中カテゴリ“ファイルマネージャー”や“デスクトップ”、“コンテキストメニュー”では、“ファイルの実行”や“ファイル名の変更”といったマルウェア実行時に必要なファイル操作に起因するコードが多く抽出された。マルウェア実行には、ファイル名や拡張子の認識、リネーム処理などが前提となる場合が多く、自動化のためにはファイル属性の正確な把握が不可欠であると推察される。

中カテゴリ“コマンドライン”では、“コマンドの実行”や“一時停止の解除”，“ファイルの実行”など、システム制御からファイル操作に至る幅広い操作目的が確認された。確認された事例は、RATによる攻撃活動を模倣したコマンド入力画面であり、プロセスの強制終了や外部サーバとの通信確立などのコマンドを、解析環境で入力することで、マルウェアの挙動を再現できるものである。このようなコマンドライン操作では、単純なキー入力だけでなく、コマンド履歴やエラーメッセージの確認、実行結果の成否判断といった多様なテキスト情報の認識が不可欠であり、正確な解析にはそれらを適切に理解する能力が求められる。

Web ブラウザ内で確認される中カテゴリ“ブラウザページ”や“ブラウザメニュー”では、“ファイルのダウンロード”，“CAPTCHA 認証の通過”や“Clickfix の操作実行”といったブラウザを介したマルウェアの取得過程での操作目的が確認された。したがって、URL を起点とする動的解析では、これらマルウェア取得に不可欠な Web ページ上

での適切な操作の自動化への対応が重要と再認識した。

中カテゴリ “メール” では、他のイベントタイプと比較して操作のバリエーションが少なく、“添付ファイルの開封・実行” に限定された。そのため、自動化においては、LLM がメール本文や GUI 上でファイルが添付されていることを正確に認識できるかが重要と推察される。

中カテゴリ “マクロ付きファイル” では、主な操作として “マクロの有効化・実行” および “ファイルのダウンロード” が確認された。中には、マクロ有効化・実行を促す文章やダウンロードリンクがスクロール後に表示される位置に含まれる事例も確認された。したがって、マクロ付きファイルの操作を自動化するには、ファイル全体の内容を把握した上で、マクロやリンクに関連する文章を抽出し理解する能力が求められる。

中カテゴリ “解析用アプリ”, “マルウェア関連アプリ”, “ファイル名を指定して実行”, および “アプリ” では、他のイベントタイプと比較しても多様な操作目的が存在することが確認される。これらのイベントタイプでは、ボタンやメニューの配置、入力欄の有無、画面遷移の仕様などアプリごとに異なるため、自動化のためには GUI 構造や表示内容を正確に認識し、適切な操作を選択する必要がある。

大カテゴリ：解析操作

大カテゴリ “解析操作” では、イベントに対して解析進行のために著者らが必要と判断した解析操作の種類を中心カテゴリとして抽出し、各操作における変数をカテゴリとしてコードを抽出した。中カテゴリで抽出された解析操作は、“クリック”, “ダブルクリック”, “右クリック”, “キー入力”, “ホットキー”, “スクロール” の 6 種類である。クリック系の操作では、イベント上の “ボタン” や “メニューリスト” に含まれるラベルを認識して適切な対応が必要であることが確認された。キー入力では、“パスワード” や “CAPTCHA の文字列” など、イベントの情報を正確に認識した上で入力内容を決定する必要があることが確認された。さらに、ホットキー入力では、“ペースト (Ctrl+V)” などのテキスト編集操作や、“ファイル名を指定して実行 (Windows+R)” といった OS 固有の機能を呼び出すための操作が確認された。スクロール操作では、ページ全体の情報を把握するための下方向へのページ移動操作が確認された。自動化のためには、特定された操作方法および操作に伴う変数をどちらも正確に抽出する必要がある。

5. Stage2 : LLM の活用可能性の検証

本研究では、Stage1 で抽出したイベントと解析操作を基にデータセットを作成し、そのデータセットで LLM の活用可能性を検証した。まず、Stage1 で抽出したイベントと解析操作を紐づけたイベントデータセットを作成した(イベントデータセットの作成)。次に、データセット内のイベント画面と作成したプロンプトを LLM に入力し、解析操

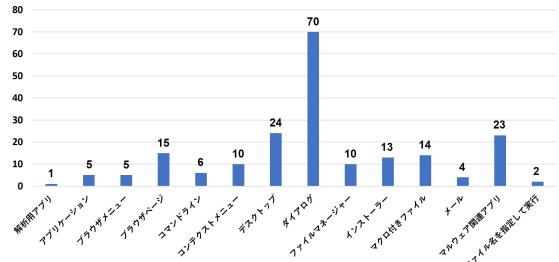


図 2 イベントデータセットに含まれるイベントタイプの内訳

作を抽出した(解析操作の抽出)。その後、LLM により抽出された解析操作とデータセット内の著者らにより抽出された解析操作との一致度を評価した(解析操作の一一致度評価)。以降の各項では、Stage2 の各手順を詳述した後、検証結果を示す。

5.1 イベントデータセットの作成

Stage1 で抽出したイベントと解析操作を紐づけ、Stage2 の検証のためのイベントデータセットを作成した。このデータセットは、コーディングにより抽出された延べ 508 件のイベントから、重複を除いた 202 件のイベント画面とそれに対応する解析操作で構成される。データセットに含まれるイベントタイプの内訳を図 2 に示す。

5.2 解析操作の抽出

本研究では、他タスクでの先行研究 [11], [12] で性能の高さが示唆される OpenAI の GPT4.1(gpt-4.1-2025-04-14), GPT-4o(gpt-4o-2024-11-20), GPT-4-turbo(gpt-4-turbo-2024-04-09)^{*3} の 3 種類の LLM を API 経由で利用して解析操作を抽出する。また、LLM の出力のランダム性を調整する temperature は高いほど多様で創造的になる一方で不安定さも増し、低いほど安定して一意になる傾向にあるため、本研究では先行研究 [11] に従って 0.25 に設定することで、創造性を残しつつ過度なランダム性を抑制した。なお、いずれの LLM もファインチューニング等は行わないものとする。本研究では、データセット内のイベント画面と、先行研究 [8], [12] を参考に作成した表 2 に示すプロンプトを入力とした。プロンプトには、役割の設定や 4.3 節で特定された解析操作に基づく出力形式の指定に関する文言を導入した。

5.3 解析操作の一一致度評価

ある 1 枚のイベント画面について、LLM により抽出された解析操作列 $A = [A_1, A_2, \dots, A_n]$ と、著者によるコーディングで抽出された解析操作列 $\hat{A} = [\hat{A}_1, \hat{A}_2, \dots, \hat{A}_m]$ とし、 A と \hat{A} の一致度を算出する。評価指標は、先行研究 [11] を参考に以下で定義される正解率を用いる。

^{*3} <https://platform.openai.com/docs/models>

表 2 本検証で設定したプロンプト全文

Prompt

あなたはマルウェア解析者です。添付した画像について以下の質問に回答してください。ここで、イベントとは動的解析中に解析環境内で表示され、解析を進行させるためにユーザ操作を要する GUI と定義します。

Q1：添付された画像に表示されているイベントに対して、マルウェア解析を進行させるためにはどのような操作を実施すべきですか。最も適切な操作内容を出力してください。ただし、操作後の画面遷移は考慮せず、添付した画像内での操作内容を出力してください。出力形式は（行動：クリック、実行箇所：）、（行動：ダブルクリック、実行箇所：）、（行動：右クリック、実行箇所：）、（行動：キー入力、キー：）、（行動：ホットキー、キー：）、（行動：スクロール、方向：）のいずれかのみとし、最も解析に適した操作を回答してください。変数部分はイベントを基に解析に適した内容を記載してください。複数の手順を要する場合は、その手順を時系列順にすべて記述してください。

Q2：回答した根拠を簡潔に記述してください。出力にはイベントの内容、操作の目的を含めてください。

表 3 イベントタイプごとの解析操作の一一致度の平均 Acc.[%]

イベントタイプ	GPT-4.1	GPT-4o	GPT-4-turbo	平均
解析用アプリ	50.00	50.00	0.00	33.33
アプリ	44.00	34.00	60.00	46.00
ブラウザメニュー	80.00	100.00	60.00	80.00
ブラウザページ	84.44	75.55	73.33	77.77
コマンドライン	47.62	57.14	52.38	52.38
コンテキストメニュー	30.00	20.00	0.00	16.67
デスクトップ	56.25	61.11	53.47	56.94
ダイアログ	96.43	84.64	88.57	89.88
ファイルマネージャー	66.67	66.67	63.33	65.56
インストーラー	92.31	92.31	88.46	91.03
マクロ付きファイル	57.14	53.57	57.14	55.95
メール	87.50	58.33	87.50	77.78
マルウェア関連アプリ	62.31	45.43	38.55	48.76
ファイル名を指定して実行	100.00	66.67	50.00	72.22
全体	68.19	61.81	55.19	61.73

$$Acc. = \frac{LCS(A, \hat{A})}{\max(|A|, |\hat{A}|)} \quad (1)$$

ここで、 $LCS(A, \hat{A})$ は解析操作列 A と \hat{A} の最長共通部分列の長さ、 $|A|$ は LLM が output した解析操作のステップ数、 $|\hat{A}|$ は著者による解析操作の正解ステップ数を表す。LCS を採用した理由は、順序を保持したまま共通部分列を抽出できるという特性にあり、操作列中に余分な操作が含まれている場合や、一部の操作のみが正しく実行されている場合においても、正確に評価するためである。また、分子に $\max(|A|, |\hat{A}|)$ を用いることで、操作ステップ数の過不足に対しても適切にペナルティを与えて評価する。

5.4 RQ2：結果

各イベント画像に対して式 (1) で一致度を算出した後に、イベントタイプごとに平均を算出した結果を表 3 に示す。本評価において、GPT-4.1 は 68.19% と最も高い精度を示し、次いで GPT-4o (61.81%)、GPT-4-turbo (52.81%) の順となった。GPT-4.1 は 2025 年にリリースされ、3 種類の中で最も新しいモデルであり、他モデルと比較して正確

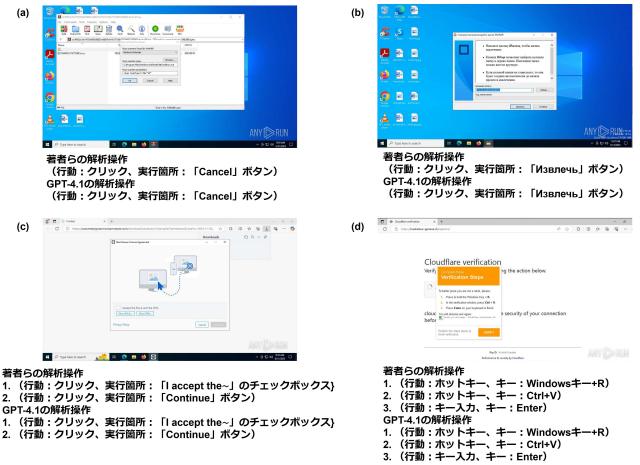


図 3 LLM が抽出した解析操作例

な解析操作の抽出が可能であることが示唆された。特に、“ダイアログ” (96.43%) や “インストーラー” (92.31%) における適切なボタンクリック操作において優れた性能を発揮した。GPT-4o は “ブラウザメニュー” (100.00%) や “インストーラー” (92.31%) など一部のイベントタイプで GPT-4.1 に匹敵する精度を示した。GPT-4-turbo は全体的に他モデルより精度が劣り、特に “コンテキストメニュー” (0.00%) などで著しく低い結果となった。

イベントタイプに着目すると、“インストーラー” (91.03%)，“ダイアログ” (89.88%) など、ボタンやメニュー等の明確な UI 要素のクリック操作を伴うイベントタイプでは全モデルが高い精度を示す傾向にあった。一方、“コンテキストメニュー” (16.67%) は全モデルで 30% 以下であり、最も低い精度を示した。また、“解析用アプリ” (33.33%)，“アプリ” (46.00%)，“マルウェア関連アプリ” (48.76%) など、アプリ特有の情報の認識が求められる場合、精度が低い傾向にあることが確認された。

次に、高い精度を示した GPT-4.1 で正しく解析操作が抽出された、図 3 に示すイベントと解析操作例に着目し、LLM が対応可能な事例を言及する。(a) は、ウイルススキャンの実行要否を確認する “ダイアログ” である。LLM は、スキャンを実行するとマルウェアが削除・隔離される可能性があるため、“Cancel” をクリックしてダイアログを閉じる必要があると判断し、内容を正しく認識した上で適切な操作を抽出した。(b) は、ロシア語の記載がある WinRAR (“ファイルマネージャー”) である。LLM は、“И з в л е ч ь (抽出)” ボタンを押すと展開が開始されると判断し、正確にロシア語を認識してクリックすべきボタンを抽出した。また、中国語の記載があるイベントでも正しく認識できる事例が確認された。(c) は、“ライセンスの同意” を要求する “アプリ” である。LLM は、チェックボックスを選択しないと “Continue” ボタンが有効化されず処理が進まないと認識し、同意チェックとボタンクリックの二段階の操作を手順通り抽出した。(d) は、“Clickfix

の操作実行”が求められる“ブラウザページ”である。LLMは、マルウェアのダウンロードや実行を誘導する典型的な手法で、解析には指示通りのキー操作が必要と判断し、攻撃手法の文脈を踏まえて解析操作を抽出した。

6. 議論

5.4節で示したとおり、本研究で用いたモデルとプロンプトによる一致度は最大68.19%にとどまり、動的解析を完全に自動化するために必要な操作を十分に抽出できたとは言い難い。一方で、従来手法では困難であったイベントへの対応が示唆されたことから、LLMによるイベント認識と解析操作抽出は今後の動的解析自動化の性能向上に寄与し得ると考えられる。また、本研究で用いたプロンプト設計は単純であるため、得られた知見を踏まえた改良の余地が大きい。6.1節では、従来手法では対応が困難である事例を挙げ、従来手法の限界に対するLLMの有効性について分析する。6.2節では、本検証の失敗事例を挙げ、今後のプロンプト設計等の改良に向けた指針を示す。最後に本研究の制約と倫理的配慮について述べる。

6.1 従来手法の限界に対するLLMの有効性

ダイアログへの柔軟な対応

第一に、ダイアログに関する従来のキーワードマッチング手法では対応できないイベントにおいて、LLMは適切に応答できる点である。例えば、図3(a)では従来手法のキーワードリストが“OK”や“Next”といった肯定的なワードで構成されるため、“Cancel”的クリックを要求するイベントには対応できない。また、単純に“Cancel”をリストに追加しても、(a)のように“OK”と“Cancel”が並存する場合は正しい選択ができない。これに対しLLMは、ダイアログの内容を踏まえた判断により適切な操作を抽出できた。また(b)において、従来手法では多言語のキーワードをリストに設定すれば対応可能と推察されるが、すべての言語を網羅的に設定することは現実的ではない。一方で、LLMは多言語に対応するための追加学習無しで、多言語のイベントを理解し、適切な解析操作を抽出できた。

多様かつ複雑な操作を要するイベントへの対応

第二に、ダイアログ以外の多様かつ複雑な操作を要するイベントに対しても、LLMは適切に応答できる点である。例えば、コマンドラインのように入力を伴うイベントにおいても応答が可能であり、これはRATやマルウェアビルダの動作の自動進行に関わるため、解析者にとって有用である。本研究の結果から、従来手法では対応しきれていないコマンドラインへの操作一致度が約50%に達することが確認された。さらに、マルウェア解析に適したプロンプト設計を工夫すればさらなる精度向上が見込まれる。また、他のイベントタイプにおいても、LLMは複雑な操作の応答が可能であることが確認された。例えば、(c)では、複数

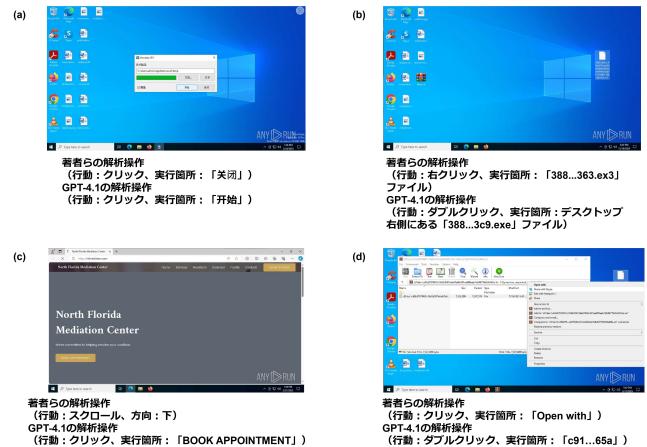


図4 LLMが誤った解析操作を抽出したイベント例

のUI要素間の依存関係や段階的な操作要求といったルールベースでは扱えない複雑なイベントにおいても、LLMは適切な解析操作を抽出できた。また、(d)のClickfixでは、LLMはテキストや攻撃的意図を総合的に解釈し、適切な解析操作を手順通り抽出できた。さらに、Clickfixのようにユーザ操作を契機とする攻撃手法は、Filefix [15]のような新しい手法へと発展しているため、LLMはこのような未知の攻撃にも柔軟に対応できる可能性があるといえる。

6.2 LLMの活用に際しての課題と今後の研究可能性

本節では、失敗事例から課題を抽出し、プロンプト設計の工夫を中心に今後の研究可能性を議論する。ここでは、3種類のLLMによる全28件の失敗事例に着目し、代表的な失敗要因を3つに分類した。失敗事例を図4に示す。

イベント内容の認識不足

イベント画面の情報を正しく認識できず、LLMが誤った操作を抽出する事例が確認された。例えば、圧縮ファイルの展開に関する図4(a)では、著者は進捗バーが右端まで到達していることから処理完了と認識し、次工程に移るためアプリを閉じる操作が必要と判断した。一方で、LLMは進捗バーを認識せずに、展開処理を実行する必要があると判断し、展開開始の操作を抽出した。また、“デスクトップ”的(b)では、著者はファイルの拡張子が“ex3”であるため実行前にRename操作が必要と判断した。一方で、LLMは、“.exe”と誤認し、ファイル実行のためダブルクリック操作を抽出した。このような認識の齟齬は、LLMが進捗バーや拡張子といった視覚的・属性的情報を十分に捉えられていないことに起因する。この問題に対処するためには、進捗バーの充足度やファイル名・拡張子の正確に抽出させるような着目点の明示的な指示を含むプロンプト設計が有効と考えられる。さらに今後は、解析に必要な要素を自動で抽出・解釈できるプロンプト設計が求められる。

解析操作の優先度の違い

著者らとLLM間の解析操作の優先度の違いによりLLM

が誤った事例が確認された。例えば，“ブラウザページ”の図 4(c)では、著者らは“ページ全体の確認”を優先し、スクロール操作が必要と判断したが、LLM は、ボタン押下で次の挙動が確認できると判断し、ボタンへの即時操作を優先した。このような優先度の違いは、LLM がイベント全体の確認や解析処理の流れを十分に考慮できていないことに起因する。今後は、明示的なボタンやリンクが表示される場合でも、他に重要な操作指示がないかページ全体の確認を優先するなど、操作の優先順位付けに関する知識を LLM に与えることが有効と考えられる。

目的達成のための解析操作の違い

著者らと LLM 間で解析操作の目的は一致するが、目的達成方法の違いにより LLM が誤った事例が確認された。例えば、WinRAR 上で表示されるコンテキストメニューの図 4(d)では、著者らは“ファイルの実行”を目的に“open with”を選択した。一方で、LLM はコンテキストメニューを認識しつつファイル実行のためにダブルクリック操作を抽出した。操作の違いに優劣はないが、意図しない挙動を防ぐには、操作判断基準を明示するプロンプト設計や過去の成功事例を参照する仕組みの導入が有効と考えられる。

6.3 制約

本研究では ANY.RUN のイベント画面を対象にコーディングした。このため、文献 [16] 等と同様に完全性を主張するものではない。その代わりに、理論的飽和の概念 [14] に従い、新コードが発見されるまでコーディングを反復し、より多くのイベント抽出に努めた。さらに、Phase1 ではマルウェアファミリに基づくレポート選定を行ったが、標的型攻撃に利用されるマルウェアが含まれない可能性を懸念し、Phase2 では別途検索ワードを設定し、異なる観点からのレポート選定により調査の網羅性を高めるよう努めた。

6.4 倫理的配慮

本研究は研究倫理に関するメンロレポート [17] に準拠して設計された。ANY.RUN のイベント画面には、メールアドレス等の個人情報が含まれる場合があったため、慎重に確認した上で本稿にイベント画面を掲載した。また、イベント画面の収集は、ANY.RUN の規約で自動収集するクーラの使用が禁止されるため、すべて手作業で実施した。

7. おわりに

本稿は、動的解析中のイベントに対する自動応答への LLM の活用可能性を検証することを目的とした。コーディングを通してイベント・解析操作を抽出するとともに、LLM がイベントから適切な解析操作の抽出に資する性能を有することを示した。特に、本研究により、ルールベースの従来手法では対応困難な Clickfix 等の複雑なイベントに LLM が対応できることが確認された。加えて、精度向上

に向け、LLM の課題を抽出し、プロンプト設計を中心に研究可能性を言及した。今後は、プロンプト設計の工夫や実際に LLM を活用した自動応答システムの開発を検討する。

謝辞 本研究は日立グループ内のサイバー攻撃解析に関わる佐藤隆行氏や専門家各位に有益な助言とご協力を頂きました。深く感謝致します。

参考文献

- [1] Wong, M., Y., et al.: An inside look into the practice of malware analysis, CCS '21, pp. 3053–3069 (2021).
- [2] Kaspersky Lab: ClickFix technique: what it is and why it's dangerous, available from <https://www.kaspersky.com/blog/what-is-clickfix/53348/> (2025-08-06 accessed).
- [3] Lengyel, T., K., et al.: Scalability, Fidelity and Stealth in the DRAKVUF Dynamic Malware Analysis System. ACSAC '14, pp. 386–395 (2014).
- [4] ANYRUN: ANYRUN, available from <https://any.run/> (2025-07-03 accessed).
- [5] Joe security LLC: JoeSandbox cloud basic, available from <https://www.joesandbox.com/> (2025-07-03 accessed).
- [6] Gruber, J. et al.: Fighting Evasive Malware How to Pass the Reverse Turing Test By Utilizing a VMI-Based Human Interaction Simulator. Datenschutz und Datensicherheit(DuD) '22, pp. 284–290 (2022).
- [7] Joe security LLC: LEVEL UP: Detecting Phishing with GenAI, available from <https://joesecurity.org/blog/6811663389969520216> (2025-07-03 accessed).
- [8] Fujii, S., et al.: Feasibility Study for Supporting Static Malware Analysis Using LLM. SECAI '24, pp. 5–28 (2024).
- [9] Wong, M., Y., et al.: Understanding LLMs Ability to Aid Malware Analysts in Bypassing Evasion Techniques. ICMI Companion '24, pp. 36–40 (2024).
- [10] Zhuosheng, Z., et al.: You Only Look at Screens: Multi-modal Chain-of-Action Agents. ACL '24, pp. 3132–3149 (2024).
- [11] Wen, H., et al.: AutoDroid: LLM-powered Task Automation in Android. ACM MobiCom '24, pp. 543–557 (2024).
- [12] Gao, D., et al.: ASSISTGUI: Task-Oriented Desktop Graphical User Interface Automation. CVPR '24, pp. 13289–13298 (2024).
- [13] ANYRUN: Malware Trends Tracker, available from <https://any.run/malware-trends/> (2025-06-01 accessed).
- [14] Glaser, B., et al.: The Discovery of Grounded Theory: Strategies for Qualitative Research. (1967).
- [15] BleepingComputer: New FileFix attack weaponizes Windows File Explorer for stealthy commands, available from <https://www.bleepingcomputer.com/news/security/filefix-attack-weaponizes-windows-file-explorer-for-stealthy-powershell-commands/> (2025-08-06 accessed).
- [16] Wong, M., Y., et al.: Comparing Malware Evasion Theory with Practice: Results from Interviews with Expert Analysts, SOUPS '24, pp. 61–80 (2024).
- [17] Bailey, M., et al.: The Menlo Report, Technical report, U.S. Department of Homeland Security, (2012).