КІБЕРБЕЗПЕКА: освіта, наука, техніка
CYBERSECURITY: EDUCATION, SCIENCE, TECHNIQUE
№ 3 (27), 2025
ISSN 2663 - 4023
DOI 10.28925/2663-4023.2025.27.746
УДК 004.056

Anatolii Chornyi
Master's Degree in Cyber Security
National Technical University of Ukraine
"Igor Sikorsky Kyiv Polytechnic Institute", Kyiv, Ukraine
ORCID ID: 0009-0001-4147-9084
anacho-ipt23@lll.kpi.ua

Iryna Stopochkina
PhD, Associate Professor
Associate Professor at the Education and Scientific Institute of Physics and Technology
National Technical University of Ukraine
"Igor Sikorsky Kyiv Polytechnic Institute", Kyiv, Ukraine
ORCID ID: 0000-0002-0346-0390
i.stopochkina@kpi.ua

GRAPH-BASED ANALYSIS OF INFORMATION FLOWS IN TELEGRAM FOR CYBERSECURITY THREAT DETECTION

Abstract. This paper explores modern methods for analyzing information flows in messengers, emphasizing their role in cybersecurity. The study compares different approaches, including API-based data collection, the use of graph and relational databases, and the automation of open data gathering. Special attention is given to the theoretical foundations of information flow analysis, focusing on the social graph concept and its application in modeling the dissemination of information across networks. The advantages of graph databases for detecting, visualizing, and analyzing networks of information distribution are examined, highlighting their effectiveness in uncovering hidden connections between channels. A prototype system for automating open data collection has been developed, integrating methods for extracting, processing, and structuring information from messenger platforms. The proposed system employs a combination of graph-based and relational techniques to enhance the accuracy and efficiency of detecting interconnections between communication channels. A series of computational experiments has been conducted to validate the effectiveness of the developed algorithms and software prototypes. The results confirm that combining these methods significantly improves the ability to identify information threats, including disinformation campaigns, automated bot activity, and coordinated attacks within messenger ecosystems. Actionable recommendations for the practical implementation of these approaches in cybersecurity tasks are provided. Specifically, they outline strategies for improving the monitoring and detection of malicious information activities, optimizing data collection and analysis pipelines, and leveraging graph-based insights to enhance situational awareness in digital communication environments. These findings contribute to the ongoing development of advanced cybersecurity solutions aimed at mitigating risks associated with modern information warfare.

Keywords: cybersecurity; information flows; messengers; graph databases; data collection automation.

INTRODUCTION

Problem Statement. Messengers, as one of the key channels of modern communication, are increasingly used not only for exchanging messages but also for disseminating various types of information, including harmful or manipulative content. With the growing number of disinformation campaigns aimed at undermining trust in public institutions, influencing electoral processes, or destabilizing economic conditions, identifying the sources and pathways of such information flows has become a critical cybersecurity challenge.

Existing methods for analyzing information flows in messengers do not always allow for the timely and accurate identification of information dissemination networks due to the complexity of structures and limited access to data. This necessitates the development and improvement of analytical approaches based on modern technologies such as messenger APIs, graph databases, and automated open data collection systems.

Review of Recent Studies and Publications. Recent research on social network analysis has established the foundation for studying information dissemination using graph-based methods to model connectivity and influence [1], [2]. The small-world network model has been widely applied to explain the efficiency of information spread in large digital ecosystems [3]. Identifying key players in social structures has also been a major focus, as it enables the detection of influential nodes and coordinated activity within networks [4].

Significant attention has been given to the role of misinformation and algorithmic bias in shaping online discourse. Studies highlight the large-scale spread of fake news, driven by recommendation systems that amplify certain narratives [5], [6]. Integrating content analysis with network structures has proven effective in detecting hidden relationships and addressing cybersecurity threats [7], [8]. Dynamic network analysis further enhances these capabilities by capturing evolving interactions and structural changes over time [9].

Cybersecurity research has also explored the risks associated with social networks, including the potential for adversarial use, such as exploiting network structures to maximize the impact of cyberattacks [10]. The propagation of malicious content and malware through trusted connections within digital environments presents a significant challenge, requiring advanced detection and mitigation strategies. However, existing studies lack comprehensive approaches that combine graph databases, relational databases, and automated data collection methods for real-time detection of harmful information dissemination networks. Addressing this gap is crucial for advancing monitoring techniques and strengthening information security.

Objective of the Study. The objective of this study is to develop new approaches for analyzing information flows in messengers using graph databases, automating open data collection through APIs, and designing effective methods for detecting information dissemination networks to enhance the cybersecurity of the information space.

RESEARCH METHODOLOGY

Methodology for Analyzing Information Dissemination Flows in Messengers

The analysis of information dissemination flows in messengers is conducted based on repost analysis, message text content, and message metadata (such as publication time and the geographical location of the source). The relationships between channels are represented as a graph, where the nodes correspond to channels, and the dissemination of information (connections) is represented by edges.

Graph databases are commonly used for analyzing information flows in messengers, as they enable the creation of visual models of network connections between channels. Using databases like Neo4j, graphs can be constructed where nodes represent channels or communities, and edges represent reposts or other forms of content transmission. Analyzing such graphs helps identify key points of information flow concentration and predict how information might spread further. One important metric is degree centrality, which assesses the influence of each channel in the network on message dissemination.

Applying information flow analysis through reposts allows for the detection of so-called “fake news” and manipulative campaigns. Identifying abnormal patterns, such as unusual

activity from specific channels or suspicious reposts, can indicate organized propaganda or disinformation efforts.

The use of messenger APIs for automating open data collection significantly reduces the labor intensity of the process. The Telegram API, for instance, provides access to information about channels, messages, and reposts, establishing logical connections between various sources. This opens up new possibilities for in-depth analysis, including real-time monitoring of information flows with a focus on repost activity.

Formalized Description of the Methodology in a Step-by-Step Algorithm:

1. **Initialization (Data Collection)**
   - **Condition:** Data for analysis has been collected from the messenger (e.g., Telegram).
   - **Action:** Collect open data on channels and their posts using the API.
   - **Result:** Creation of a data table $D = \{d_1, d_2, \ldots, d_n\}, where\ d_i$ represents a record of a post in a channel.
     $$D = \{(c_1, t_1, p_1), (c_2, t_2, p_2), \ldots, (c_n, t_n, p_n)\}$$
     where:
     c_i — channel;
     t_i — publication time;
     p_i — publication data.
2. **Content Analysis (Identifying Connections)**
   - **Condition:** There is a relationship between posts in different channels (reposts).
   - **Action:** Compare post texts across different channels to detect similar or identical publications.
   - **Result:** Creation of a list of relationships between channels $R = \{(c_i, c_j)\}$ where c_i and c_j are channels with identified reposts.
     $$R = \{(c_i, c_j) |\ if\ R = p_i \sim p_j\ for\ t_i, t_j \in T\}$$
     where:
     $p_i \sim p_j$ — similarity between posts in channels ci and cj;
     T— set of publication times.
3. **Creating the Information Dissemination Graph (Graph Database)**
   - **Condition:** Connections between channels have been identified.
   - **Action:** Create a graph model based on detected relationships between channels and posts.
   - **Result:** Construction of a graph where nodes represent channels $C = \{c_1, c_2, \ldots, c_n\}$, and edges represent repost relationships $E = \{(c_i, c_j)\}$.
     $$G = (C, E)$$
     where:
     C — set of channels;
     E — set of connections between channels.
4. **Identifying Dissemination Networks (Component Analysis)**
   - **Condition:** The graph has been constructed.
   - **Action:** Apply a connectivity component search algorithm (e.g., Depth-First Search or Breadth-First Search) to identify groups of channels where intensive information dissemination occurs.

- **Result:** Determination of connectivity components in the graph G.

$$Components(G) = \{C_1, C_2, \ldots, C_k\}$$

where C_k represents a connectivity component consisting of channels $c_1, c_2 \ldots c_m$ between which reposts exist.

Detection of Explicit and Implicit Connections

Explicit reposts serve as key indicators for identifying direct interactions between channels when one channel publishes content previously posted by another. This creates clear connections that help in constructing the primary interaction graph.

Implicit connections can be identified through indirect interactions between channels, such as shared users, domains, or hashtags. Detecting such connections allows the identification of networks of channels that may not be immediately apparent but can significantly influence the spread of information within the network. This requires analyzing not only explicit reposts but also common elements that may indicate implicit connections between channels.

System Architecture for Detecting Information Dissemination Flows in Messengers

The system architecture for detecting information dissemination flows in messengers (Fig. 1) is crucial for effectively analyzing large volumes of data and ensuring accurate results. It consists of several core components, each playing a vital role in data collection, processing, and visualization.

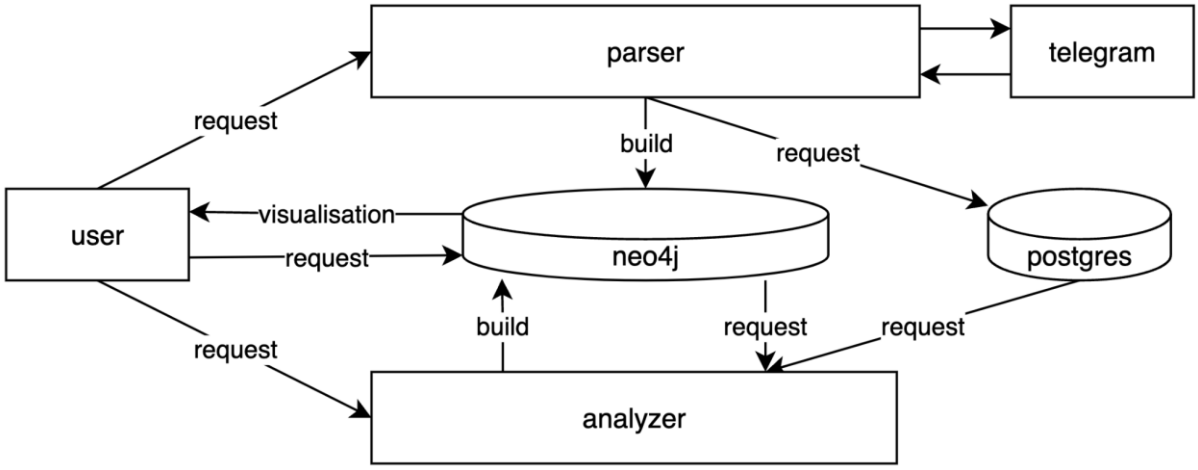

Fig. 1. System Architecture

Key Components of the Architecture:

1. User

The user interacts with the system by submitting queries to request information about content dissemination in messengers. They receive visualized data from the Neo4j graph database, which enables them to observe how information spreads between different channels, users, and other system elements.

2. Parser

The parser is a core component responsible for collecting data from Telegram via the API. It interacts with the platform, retrieves messages, and stores them in the PostgreSQL relational database for further analysis. Additionally, the parser constructs a data graph in Neo4j using collected information about channels, users, bots, and other message attributes.

3. Neo4j (Graph Database)

The Neo4j graph database is used to store and construct the information flow graph. It processes queries from users and the analyzer while also providing a visualization of information connections. Neo4j illustrates how information propagates among various elements, such as channels, domains, hashtags, and other attributes.

4. PostgreSQL (Relational Database)

The PostgreSQL relational database stores structured data collected by the parser. It is used for additional analysis, providing access to the necessary information for creating more complex models of information dissemination.

5. Analyzer

The analyzer processes data obtained from Neo4j and PostgreSQL to build models of information dissemination. It uses this data to generate insights, which are then provided to the user for deeper analysis.

Interaction Between Components

- The **user** sends queries to **Neo4j** and the **Analyzer** to retrieve the necessary information.
- The **user** receives the visualization of the information flow graph from **Neo4j**.
- The **Parser** retrieves data from **Telegram** via the API and stores it in both **PostgreSQL** and **Neo4j**.
- The **Analyzer** uses data from both **PostgreSQL** and **Neo4j** to construct models of information dissemination.

Data Preprocessing and Graph Construction Algorithm

Data Preprocessing

Data preprocessing involves several key steps, including text normalization, extraction of key attributes that will form the foundation for the graph, and filtering out irrelevant information. The primary objective is to create interconnected elements that will be represented as nodes in the graph.

Extraction of Key Attributes for Graph Construction

One important aspect is the identification of key attributes necessary for graph construction. The main attributes used to create nodes in the graph include:

- **Channels**: Channels serve as the primary units for building the graph as they are the sources of information. Defining and identifying a channel is a critical step as it creates a node in the graph that corresponds to the source of publications.
- **Users**: Users play an important role in detecting interactions between different channels. If a particular user is frequently mentioned across several channels, this helps in establishing connections between those channels, which is crucial for identifying their networks.
- **Bots**: Links to bots can be extracted using regular expressions and are represented as separate nodes in the graph.
- **Domains with Links**: URLs to external resources identify new nodes in the form of domains, enabling the analysis of interactions through external web resources.
- **Hashtags**: Hashtags act as important markers of content and themes within messages, allowing for the identification of distinct groups of channels and users.

Graph Model Construction

After data preprocessing, the next step is graph construction. This includes creating nodes for each element (channel, user, bot, domain, hashtag) and establishing relationships between them. The main relationships include:

- **REPOSTED**: This is the relationship between channels, indicating that one channel reposted a message from another channel.
- **REFERENCES**: This is the relationship between channels and domains if the messages contain links to external resources.

These relationships help to represent the network of interactions between channels and other elements, which is important for analyzing information flows.

Algorithm Optimization

An important phase in system development is optimizing the analysis algorithm, especially when working with large datasets. Key aspects include:

- **Reducing System Load**: Preprocessing methods are employed to filter out unnecessary data, minimizing the number of nodes and relationships in the graph.
- **Indexing and Efficient Queries**: This enhances the speed of data access, which is crucial for the efficient analysis of large amounts of information.

Dataset

The dataset consists of live data obtained through the Telegram API and updated in real time. It includes the following features

Table 1

Dataset Features

Feature	Description
channel_id	A unique identifier for the channel from which the message originated.
post_id	A unique identifier for the message
from_id	The identifier of the user or bot who posted the message
post_date	The date and time the message was posted
post_author	The name or alias of the message author
post_text	The text content of the message
media1_id	The identifier of the first media file
media1_size	The size of the first media file
media1_weight	The weight of the first media file
media1_height	The height of the first media file (for images)
media1_duration	The duration of the first media file (for videos)
media1_filename	The filename of the first media file
media2_id	The identifier of the second media file

media2_size	The size of the second media file
media2_weight	The weight of the second media file
media2_height	The height of the second media file (for images)
media2_duration	The duration of the second media file (for videos)
media2_filename	The filename of the second media file
fuzz_ratio_1	The fuzz algorithm discrepancy for the first comparison
fuzz_ratio_2	The fuzz algorithm discrepancy for the second comparison
fuzz_ratio_3	The fuzz algorithm discrepancy for the third comparison
fuzz_ratio_4	The fuzz algorithm discrepancy for the fourth comparison
entity_type	The type of entity (channel, user, bot, etc.)

These features enable the detailed analysis of interactions, including text and media content, and facilitate the identification of patterns and relationships between different entities within the system.

RESULTS OF THE STUDY

Analysis of the Obtained Results and Identification of Information Dissemination Networks

The results of the study allow the identification of the structure of information dissemination on Telegram through graph analysis of reposts and associated methods for detecting implicit connections. It was found that reposts in messengers are not merely isolated acts of interaction between users, but a mechanism for the circulation of information between channels and communities that form complex information networks.

Visual Analysis of the Constructed Graph

The visual analysis of the constructed graph involves examining its structure, key nodes, and connections between them to identify patterns of information dissemination in messengers. The analysis is performed using specialized tools for working with graph databases, particularly Neo4j, which allows the visualization of interconnections between channels and the identification of key points for content distribution.

The dynamics of reposts are analyzed in a temporal aspect to determine the speed of message dissemination between channels. Identifying the initial sources of information and analyzing how it spreads across the network helps to understand the features of information campaigns, including potential attempts at manipulation or coordinated dissemination.

The evaluation of the graph structure also allows the identification of possible latent connections between channels. Even if two channels do not interact directly but share common repost sources, this may indicate implicit network interactions. Using algorithms for detecting implicit connections helps uncover such indirect interactions and draw conclusions about the overall architecture of information dissemination in Telegram.

ISSN 2663 - 4023

Based on the obtained data, conclusions are drawn about the structure of information flows in the messenger, potential cybersecurity threats are identified, and measures for their neutralization are proposed.

Identification of Implicit Connections Between Channels

The detection of implicit connections between channels on Telegram is carried out using several algorithms, each analyzing different aspects of interaction between channels, including shared content, media files, text messages, and the structure of relationships.

One of the methods involves analyzing the **file names** (Fig. 2) that are published across different channels. The algorithm collects information about the files, excluding common names such as “IMG_”, and identifies those that have been shared across multiple channels simultaneously. This allows for the identification of groups of channels that share the same files and the creation of connections between them based on this commonality.

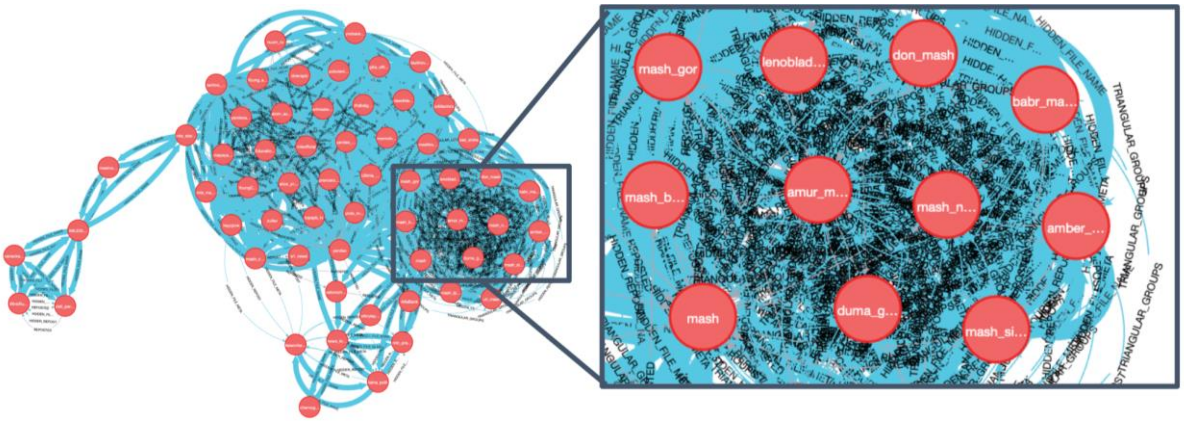

Fig. 2. Visualization of implicit connections based on file names

Another approach focuses on the **characteristics of media files** (Fig. 3), including size, weight, height, or duration. By identifying media files with identical parameters, the algorithm finds channels that use the same visual or audiovisual content. This allows for the detection of coordinated information dissemination through the publication of identical materials across different points of the network.

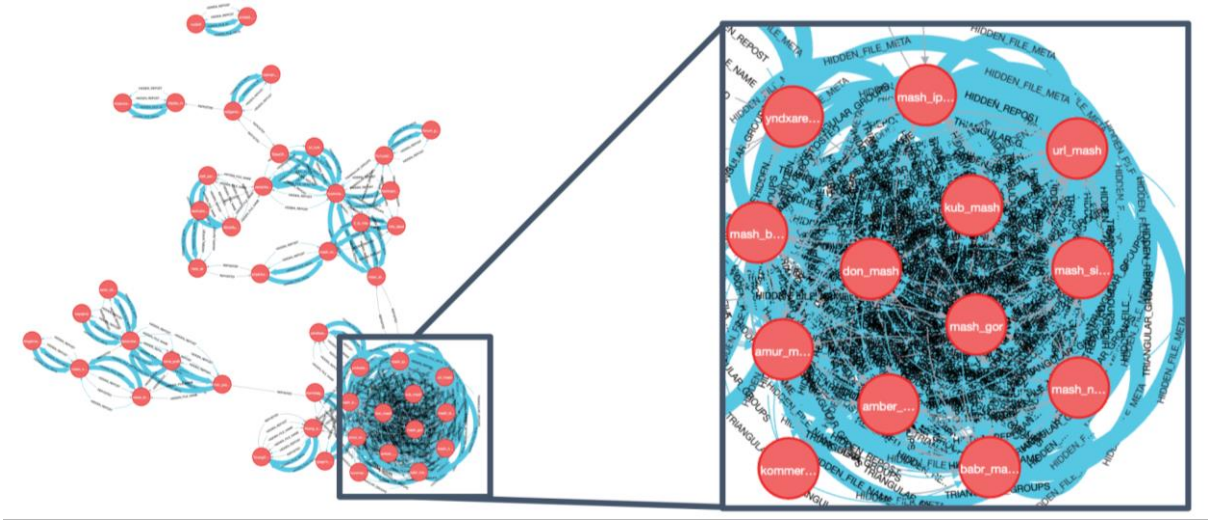

Fig. 3. Visualization of implicit connections based on files characteristics

ISSN 2663 - 4023

Text message analysis (Fig. 4) relies on the use of similarity metrics (fuzz-metrics). Messages with high textual similarity are grouped together if they appear on different channels. An important parameter is the similarity in text length within an acceptable deviation, which helps avoid false matches and focuses on truly meaningful relationships.

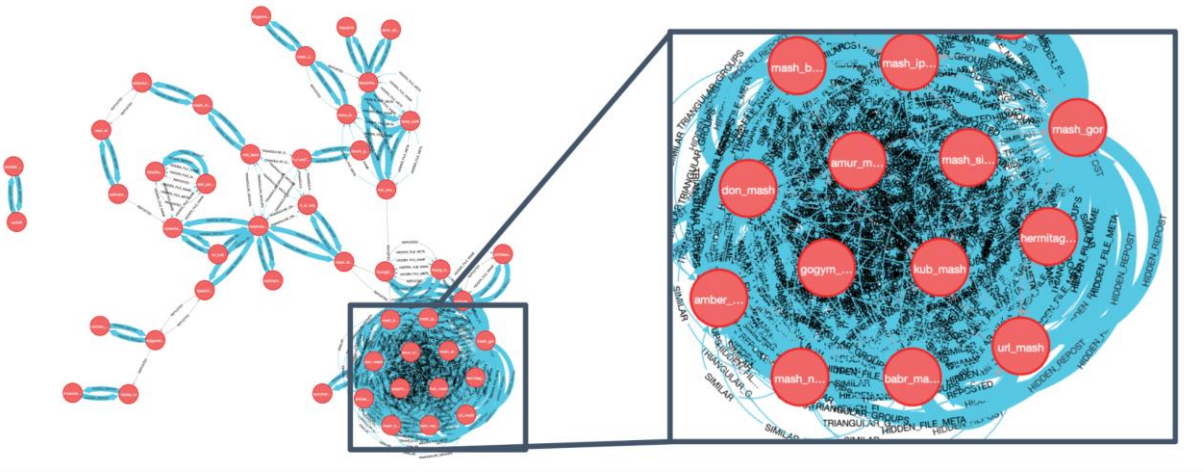

Fig. 4. Visualization of implicit connections based on text message analysis

Another approach involves analyzing the **structural interaction** (Fig. 5) between channels by identifying triangular relationships. The algorithm first finds groups of three channels interacting with each other, and then builds more complex networks by merging such triangles into larger groups. This helps to uncover more intricate interaction patterns between channels and the formation of information clusters.

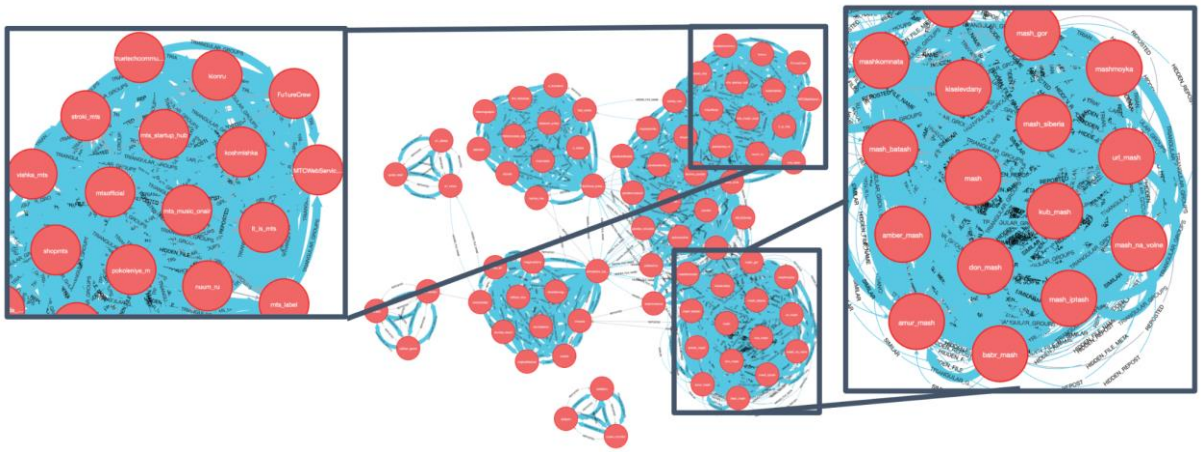

Fig. 5. Visualization of implicit connections based on structural interaction

Together, these methods provide a detailed picture of information dissemination in Telegram, reveal implicit relationships between channels, and assess potential threats to information security.

Interpretation of the Results

The interpretation of the results is based on the analysis of the constructed graph model, which reflects the network of interactions between Telegram channels based on various parameters, including shared content, media files, textual matches, and structural relationships.

The study of shared content and its dissemination showed that certain groups of channels systematically use the same materials, which may indicate shared information sources or centralized network management. The analysis of media files confirmed the existence of implicit connections between channels that, at first glance, did not exhibit common activity but published the same content with minimal modifications. The analysis of textual similarity revealed channels using similar or identical messages, suggesting potential mechanisms for automated information dissemination.

The overall interpretation of the obtained data allows for establishing the structure of information dissemination in Telegram and assessing risks associated with disinformation campaigns, bot networks, or organized groups spreading manipulative content. Identifying implicit relationships between channels opens new opportunities for monitoring the information space, which is a crucial aspect of cybersecurity and information protection.

CONCLUSIONS AND PERSPECTIVES FOR FURTHER RESEARCH

The research provides a comprehensive analysis of information dissemination networks in Telegram, focusing on the identification of implicit connections between channels through graph-based methods. The study demonstrated that reposts in messengers are not merely isolated acts of user interaction but serve as mechanisms for circulating information among channels and communities, forming complex informational networks. By leveraging advanced algorithms for analyzing shared content, media files, and textual similarities, this study has successfully uncovered previously implicit relationships between Telegram channels, which could provide insights into potential threats such as disinformation campaigns and coordinated content dissemination.

The results underscore the importance of utilizing graph databases, such as Neo4j, in analyzing large-scale messaging networks. By exploring these networks, it is possible to gain deeper insights into the structures of information flows and detect potential cybersecurity risks. The research also highlighted the significance of visualizing these networks to identify key nodes and implicit connections that may otherwise go unnoticed. Moreover, the dynamic aspect of information propagation over time and the identification of initial information sources play a crucial role in understanding the spread of messages and the potential for manipulation or coordinated distribution.

Further research could explore the integration of other data sources, such as metadata from other social platforms or real-time tracking of content interactions. Additionally, expanding the scope to include different languages and geographies would provide a more global perspective on how information spreads across various Telegram communities. Future work could also delve into the development of more refined algorithms for detecting subtle manipulations within information flows, ultimately contributing to enhancing cybersecurity measures and improving the detection of malicious activities in digital spaces.

REFERENCES (TRANSLATED AND TRANSLITERATED)

1. Wasserman, S., & Faust, K. (1994). Social network analysis in the social and behavioral sciences. *In Social network analysis: Methods and applications*, 1–27.
2. Scott, J. P. (2000). *Social network analysis: A handbook (2nd ed.).* Sage Publications.
3. Watts, D. J., & Strogatz, S. (1998). Collective dynamics of 'small-world' networks. *Nature, 393(6684),* 440–442. https://doi.org/10.1038/30918

4. Borgatti, S. P. (2006). Identifying sets of key players in a social network. *Computational & Mathematical Organization Theory*, *12*, 21–34.
5. Allen, J., Howland, B., Mobius, M., Rothschild, D., & Watts, D. J. (2020). Evaluating the fake news problem at the scale of the information ecosystem. *Science Advances*, *6(14)*, eaay3539. https://doi.org/10.1126/sciadv.aay3539
6. Bartley, N., Abeliuk, A., Ferrara, E., & Lerman, K. (2021). Auditing algorithmic bias on Twitter. *13th ACM Web Science Conference 2021*, 65–73. https://doi.org/10.1145/34475353462491
7. Campbell, W., Dagli, C. K., & Weinstein, C. J. (2013). Social network analysis with content and graphs. *Lincoln Laboratory Journal, 20,* 62–81.
8. Weinstein, C., Campbell, W. M., Delaney, B. W., & O'Leary, G. (2009). Modeling and detection techniques for counter-terror social network analysis and intent recognition. *IEEE Aerospace Conference,* 1–16.
9. Carley, K. (2003). Dynamic network analysis. *Dynamic social network modeling and analysis: Workshop summary and papers*, 133–145.
10. Yan, G., Chen, G., Eidenbenz, S., & Li, N. (2011, March). Malware propagation in online social networks: nature, dynamics, and defense implications. *6th ACM Symposium on Information, Computer and Communications Security*, 196–206.

КІБЕРБЕЗПЕКА: освіта, наука, техніка
CYBERSECURITY: EDUCATION, SCIENCE, TECHNIQUE
№ 3 (27), 2025
ISSN 2663 - 4023

Чорний Анатолій Юрійович
магістр
КПІ імені Ігоря Сікорського, Київ, Україна
ORCID ID: 0009-0001-4147-9084
anacho-ipt23@lll.kpi.ua

Стьопочкіна Ірина Валеріївна
к.т.н., доцент, доцент кафедри інформаційної безпеки
КПІ імені Ігоря Сікорського, Київ, Україна
ORCID ID: 0000-0002-0346-0390
i.stopochkina@kpi.ua

ГРАФОВИЙ АНАЛІЗ ІНФОРМАЦІЙНИХ ПОТОКІВ У TELEGRAM ДЛЯ ВИЯВЛЕННЯ ЗАГРОЗ В ЗАДАЧАХ КІБЕРБЕЗПЕКИ

Анотація. Ця робота досліджує сучасні методи аналізу інформаційних потоків у месенджерах, підкреслюючи їхню роль у забезпеченні кібербезпеки. У дослідженні порівнюються різні підходи, зокрема збирання даних за допомогою API, використання графових та реляційних баз даних, а також автоматизація збору відкритих даних. Особлива увага приділяється теоретичним засадам аналізу інформаційних потоків, зосереджуючись на концепції соціального графа та його застосуванні для моделювання поширення інформації в мережах. Розглядаються переваги графових баз даних для виявлення, візуалізації та аналізу мереж поширення інформації, що підкреслює їхню ефективність у виявленні прихованих зв'язків між каналами. Розроблено прототип системи автоматизованого збору відкритих даних, що інтегрує методи вилучення, обробки та структурування інформації з платформ месенджерів. Запропонована система поєднує графові та реляційні підходи для підвищення точності та ефективності виявлення взаємозв'язків між комунікаційними каналами. Проведено серію комп'ютерних експериментів для перевірки ефективності розроблених алгоритмів і програмних прототипів. Результати підтверджують, що комбіноване використання цих методів значно покращує можливості ідентифікації інформаційних загроз, включаючи дезінформаційні кампанії, автоматизовану активність ботів і координовані атаки в екосистемах месенджерів. Наведено практичні рекомендації щодо впровадження цих підходів у завданнях кібербезпеки. Зокрема, окреслено стратегії для покращення моніторингу та виявлення шкідливої інформаційної діяльності, оптимізації процесів збору та аналізу даних, а також використання графових методів для підвищення ситуаційної обізнаності в цифрових комунікаційних середовищах. Отримані результати сприяють подальшому розвитку сучасних рішень у сфері кібербезпеки, спрямованих на мінімізацію ризиків, пов'язаних із сучасними загрозами інформаційної війни.

Ключові слова: кібербезпека; інформаційні потоки; месенджери; графові бази даних; автоматизація збору даних.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Wasserman, S., & Faust, K. (1994). Social network analysis in the social and behavioral sciences. *In Social network analysis: Methods and applications,* 1–27.
2. Scott, J. P. (2000). *Social network analysis: A handbook (2nd ed.).* Sage Publications.
3. Watts, D. J., & Strogatz, S. (1998). Collective dynamics of 'small-world' networks. *Nature, 393(6684),* 440–442. https://doi.org/10.1038/30918
4. Borgatti, S. P. (2006). Identifying sets of key players in a social network. *Computational & Mathematical Organization Theory, 12,* 21–34.
5. Allen, J., Howland, B., Mobius, M., Rothschild, D., & Watts, D. J. (2020). Evaluating the fake news problem at the scale of the information ecosystem. *Science Advances, 6(14),* eaay3539. https://doi.org/10.1126/sciadv.aay3539
6. Bartley, N., Abeliuk, A., Ferrara, E., & Lerman, K. (2021). Auditing algorithmic bias on Twitter. *13th ACM Web Science Conference 2021,* 65–73. https://doi.org/10.1145/34475353462491

7. Campbell, W., Dagli, C. K., & Weinstein, C. J. (2013). Social network analysis with content and graphs. *Lincoln Laboratory Journal, 20,* 62–81.
8. Weinstein, C., Campbell, W. M., Delaney, B. W., & O'Leary, G. (2009). Modeling and detection techniques for counter-terror social network analysis and intent recognition. *IEEE Aerospace Conference,* 1–16.
9. Carley, K. (2003). Dynamic network analysis. *Dynamic social network modeling and analysis: Workshop summary and papers,* 133–145.
10. Yan, G., Chen, G., Eidenbenz, S., & Li, N. (2011, March). Malware propagation in online social networks: nature, dynamics, and defense implications. *6th ACM Symposium on Information, Computer and Communications Security*, 196–206.