Exploring Community Detection Algorithms for Analyzing Dark Web Networks: A Comparative Study

Abstract— **This study proposes an approach for community detection in the dark web using a combination of graph theory and machine learning techniques. The anonymous and decentralised nature of the dark web poses a challenge for identifying communities. To address this challenge, a graph representation of the dark web is constructed using data collected from various sources. Three popular community detection algorithms, namely Leiden, Louvain, and Label Propagation, are applied, and their performance is evaluated using metrics such as modularity. The evaluation of the approach on a dataset of dark web forums demonstrates its effectiveness in identifying meaningful communities. The results obtained from this approach can provide valuable insights into the organisation and structure of the dark web, which can be beneficial for law enforcement agencies and cybersecurity researchers in their efforts to combat illicit activities.**

I. INTRODUCTION

The groupings of nodes in a network that are more closely related to one another than to the rest of the network are found using community detection techniques. Community discovery algorithms come in a variety of forms, including probabilistic models, modularity-based techniques, and hierarchical clustering. The hierarchical relationships between the network's nodes are captured by hierarchical clustering methods through the construction of a dendrogram, which resembles a tree. These algorithms operate by repeatedly merging pairs of nodes or communities to create a single community that contains every node. To produce communities at various scales, the dendrogram can be sliced at different levels. The goal of modularity-based techniques is to optimise a quality function called modularity that compares the density of edges within and between communities. These algorithms operate by applying methods like greedy optimization, spectral partitioning, or simulated annealing to optimise the modularity function. Statistical techniques are used by probabilistic models to infer the network's underlying community structure. These algorithms characterise the network as a probabilistic graphical model and estimate the model's parameters and community structure using methods like maximum likelihood estimation for Bayesian inference. One of the most widely used measures for evaluating the quality of a community structure is modularity, which measures the extent to which nodes within the same community are more connected to each other than to nodes in other communities. Maximising modularity is a common objective in many community detection algorithms. Another important consideration in community detection is the resolution limit problem, which arises when the algorithm fails to detect communities at a certain scale or granularity. This can happen when the network has a hierarchical structure, with communities nested within larger communities, and the algorithm is not able to capture this nested structure. Community detection has numerous practical applications, including in social network analysis, recommendation systems, fraud detection, and more. It is also a topic of active research, and new methods and algorithms are constantly being developed to improve the accuracy and efficiency of community detection in different types of networks. Some crucial ideas and factors in community detection include the following: Communities that overlap: In some circumstances, nodes may be a part of more than one community, which gives rise to the idea of communities that overlap. Traditional community detection is a simpler challenge than overlapping community detection, which calls for several techniques and measurements. Dynamic Networks: Algorithms that can recognise both communities that remain stable over time and those that change over time are necessary for networks that evolve over time. Detecting communities that are significant and persistent can be done by using dynamic community detection algorithms to monitor changes in the network structure. Multiplex Networks: A multiplex network is any network that has numerous layers or types of connectivity. Algorithms that can recognise communities that are significant across all layers while accounting for the various forms of connectivity are needed for community recognition in multiplex networks. Scalability: Scalability becomes a key factor in community detection as networks get bigger and more complicated. Using methods like parallel computing and sampling, several algorithms are created to operate effectively on large-scale networks.

II. Literature Review

Community detection on the dark web is a challenging task due to the anonymous and secretive nature of the platform. Despite this, community detection on the dark web is an important area of research, as it can provide insights into the behavior and activities of various groups and individuals operating within the hidden corners of the internet. In recent years, there has been a growing interest in this area, and several studies have proposed various approaches and algorithms for detecting and analyzing communities on the dark web. In one such paper, the authors proposed a method based on community detection to facilitate the study of massive amounts of darknet traffic. Event coordination, such as network scans caused by botnets, was uncovered. There are three main types of events that have been automatically identified and isolated thanks to their methodology: vertical scans, horizontal scans, and targeted scans. This work is an exploratory look at how community detection algorithms might be used to analyse darknet traffic. In order to focus on the rare and potentially more interesting events for cyber-security applications, they plan to employ cutting-edge complex network approaches to filter out the usual noisy traffic [1]. In this paper, an unique strategy for constructing enhanced and filtered networks employing the benefits of LDA is proposed. This method employs network filtering, followed by SLTA, an algorithm for detecting overlapping communities in networks through topic propagation. Experiments indicate that the results achieved for locating overlapping communities using the aforementioned method are superior to those obtained using conventional methods on an LDA-filtered network. The presented methods are mostly beneficial for mining a VCoI (IslamicAwakening forum - an English-language forum accessible through the Dark Web portal), locating better communities, enhancing the comprehension of the networks, and streamlining administration. By their research, we are able to identify sub-communities of interest whose primary topic may pose a threat to national security[2]. This paper addresses the challenge of topic-based community key-member extraction using a strategy that combines text mining and social network analysis tools. This is accomplished by first employing latent Dirichlet allocation to construct two topic-based social networks in online forums: one social network geared towards the thread author point of view, and the other directed towards the general forum's replyers. Finally, using a social network constructed as a representation of the network of postings as a benchmark, topic-based important members are evaluated utilising various network analysis techniques. Tests were conducted successfully utilising an English-language forum accessible via the Dark Web gateway[3]. The authors introduce and assess a parallel community detection technique derived from the state-of-the-art Louvain modularity maximisation method in this study. The approach utilises a novel network mapping and data representation, as well as a communication runtime that is optimised for fine-grained applications operated on large-scale supercomputers [4]. In this paper, two agglomerative method-based algorithms (Louvain and Leiden) are introduced and evaluated. By comparison, the concept and benefit are summed up in detail. Lastly, the Leiden algorithm is considered to be the most recent and fastest algorithm compared to the Louvain algorithm. Future selection of the best community detection algorithms can be aided by this comparison, even though these algorithms have different definitions of community [5].The primary enhancements, extensions, and applications of LPA (Label Propagation Algorithm) are discussed in this survey. LPA has been demonstrated to be a simple and scalable algorithm.

However, several significant problems have been identified. Several variants have been proposed to address these problems, primarily by modifying the update rule and attempting to remove randomness from the algorithm. The original algorithm has also been adapted for various network types and modified to allow for different community detection schemes[6]. The authors of this paper propose an algorithm for discovering overlapping community structures in massive networks. The algorithm is based on the label propagation technique developed by Raghavan, Albert, and Kumara, but can detect overlapping communities. [7]

III. Methodology

A. Datasets and Methods

This study utilized a crawler to compile a dataset of textual content from various tags on dark web pages. The tags included header tags, body tags, meta tags, and others. The dataset contained three attributes: url, id, and origin id. The crawler was designed to collect content from different sources and tags to ensure a comprehensive and diverse dataset. The url attribute represents the URL of the page where the content was collected, the id attribute is a unique identifier for each document, and the origin id attribute denotes the source of the document. The collected dataset was used to construct a graph representation of the dark web for community detection using the Leiden, Louvain, and Label Propagation algorithms.

1) Labelling the dataset: To prepare the dataset for community detection using the Label Propagation algorithm, a labeled dataset of URLs was required. To achieve this, a supervised machine learning algorithm was employed that was trained on textual data extracted from websites spanning 16 distinct domains. The algorithm was used to assign categories to each URL in the dark web dataset. Additionally, a subset of the dataset was further classified into five broader categories, including software, marketplace, wiki, porn, and unknown. This categorization helped to provide a better understanding of the types of communities in the dark web and their distribution across different categories. The approach allowed for the generation of a comprehensive and accurate dataset for community detection in the dark web. Figure 1 provides a glimpse of the dataset.

	id	url_x	raw_x	Category	origin_id
0	1	http://zqktlwi4fecvo6r	crawled/zc	Computers and Technology	0
1	2	http://zqktlwi4fecvo6r	crawled/zc	Computers and Technology	1
2	3	http://zqktlwi4fecvo6r	crawled/zc	Computers and Technology	1
3	5	http://zqktlwi4fecvo6r	crawled/zc	Computers and Technology	1
4	8	http://zqktlwi4fecvo6r	crawled/zc	Health and Fitness	1
5	9	http://zqktlwiuavvvqqt	crawled/zc	Health and Fitness	1
6	10	http://zqktlwi4fecvo6r	crawled/zc	Computers and Technology	1
7	11	http://zqktlwi4fecvo6r	crawled/zc	Computers and Technology	1
8	12	http://zqktlwi4fecvo6r	crawled/zc	Computers and Technology	1
9	13	http://zqktlwi4fecvo6r	crawled/zc	Computers and Technology	1
10	14	http://zqktlwi4fecvo6r	crawled/zc	Computers and Technology	1
11	15	http://zqktlwi4fecvo6r	crawled/zc	Computers and Technology	1
12	17	http://msydqstlz2kzerc	crawled/m	Computers and Technology	1
13	21	http://jaz45aabn5vken	crawled/ja	Business/Corporate	1
14	22	http://zqktlwi4fecvo6r	crawled/zc	Computers and Technology	1
15	23	http://hss3uro2hsxfog	crawled/h:	Games	1
16	24	https://ssd.eff.org/	crawled/s:	Computers and Technology	1
17	25	http://he22pncoselnm	crawled/h	E-Commerce	1

Fig. 1. Dataset

2) Converting the datasets into JSON: To apply the community detection algorithms (Leiden, Louvain, and Label Propagation), a JSON object dataset was required. This dataset included a list of nodes with their respective ID, URL, and category, as well as a list of links with their source and target. This structured data format allowed the algorithms to analyse the relationships between different nodes in the network and identify the communities that existed within it.

B. Experimental Setup

The community detection was done by using three distinct algorithms on the dataset. These algorithms include the well-known Louvain Algorithm, its variant, the Leiden Algorithm, and the widely used Label Propagation Algorithm. The experiments were conducted on a computer with an Intel Core i7 processor, 8GB of RAM and the Microsoft Windows 11 operating system is used for all of the experiments. The programming language used for implementation was Python and JavaScript, and the network analysis library used was NetworkX.

C. Results and Discussion

The research utilized two datasets for evaluating community detection algorithms. One dataset was a subset of the other, with 2722 rows, while the original dataset contained 9386 rows.

1) Louvain Algorithm: The Louvain Community Detection Algorithm can be applied to the network of dark web URLs represented as a graph $G = (V, E)$, where V is the set of nodes representing URLs and E is the set of edges representing relationships between URLs. The algorithm starts with each node assigned to its own community and then iteratively optimizes the modularity of the network by merging communities based on the modularity gain of moving nodes to neighboring communities. This process continues until there is no further increase in modularity. The Louvain algorithm is an efficient and effective method for identifying communities in large networks and can provide insight into the hidden patterns and structures of the URLs on the dark web.

Algorithm 1 Louvain Community Detection Algorithm

Require: A network represented as a graph $G = (V, E)$, where V is the set of nodes and E is the set of edges.
Ensure: A community assignment for each node in the network.

```
Initialize the community assignment of each node to be its own singleton community.
Initialize the modularity gain ΔQ to be a positive value.
while ΔQ > 0 do
    Set ΔQ to be 0.
    for each node i ∈ V do
        for each community C_j that i is connected to do
            Compute the modularity gain ΔQ of adding i to C_j.
        end for
        Assign i to the community that gives the maximum modularity gain ΔQ.
    end for
    Merge the nodes in the same community to form new communities.
    Update the modularity gain ΔQ.
end while
Output the final community assignment for each node.
```

a) On Original Dataset: Figure 2 displays the results of applying the Louvain algorithm to the original dataset. The nodes in the graph represent the URLs present in the dark web, which have been clustered together based on their similarities to form distinct communities. The Louvain algorithm has identified these communities by optimizing the modularity of the network, which measures the degree of connectivity between nodes within communities relative to that between communities. The resulting clusters represent groups of URLs that are more densely connected to each other than to nodes outside of the cluster.

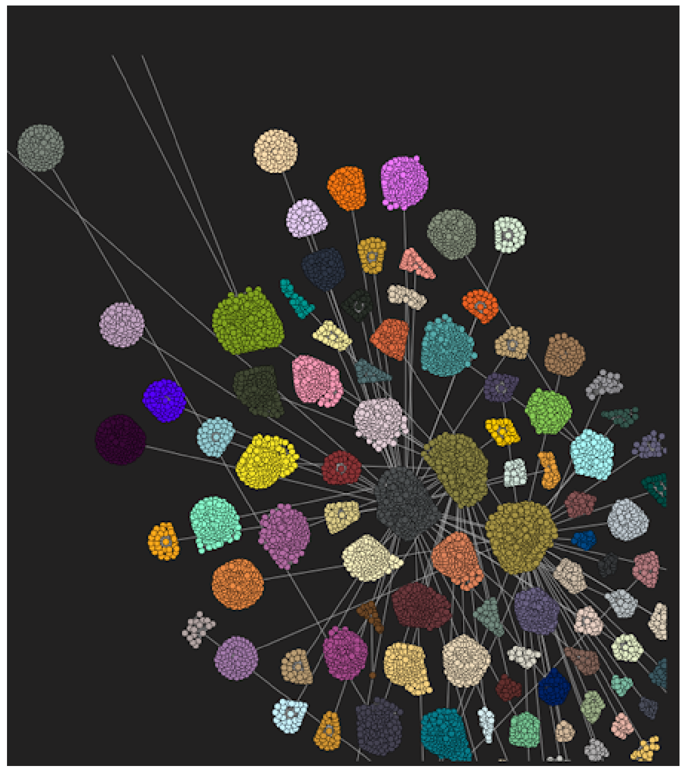

Fig. 2. Community created by Louvain Algorithm on Original Dataset

The results of community detection on the 9386 node dataset indicate that the nodes can be partitioned into 103 distinct communities. Among these communities, 92 of them have more than 50 percent of their nodes belonging to the same category. This suggests that the communities reflect strong associations between specific categories of URLs, and that many of these communities may represent subcategories or subdomains within the broader categories.

For example, one specific community with ID 99 has 267 nodes belonging to the category of 'Social Networking and Messaging', while only 1 node in the community belongs to the category of 'Computers and Technology'. This observation supports the idea that communities may represent specific subdomains within categories, as opposed to a more random grouping of nodes. Figure 3 presents the distribution of nodes in each community identified by the Louvain algorithm, with respect to the categories to which they belong. The percentage of nodes belonging to a given category is calculated for each community, and the communities are then ordered according to the category with the highest percentage of nodes.

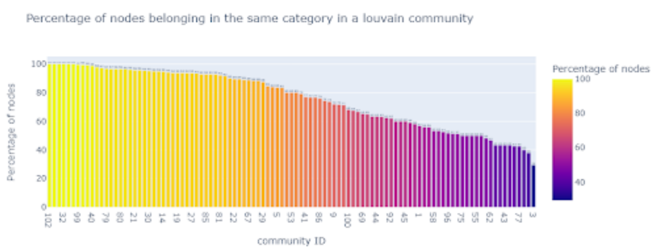

Fig. 3. Bar Chart of percent of nodes belonging to same category in a Louvain community in Original Dataset

Upon further examination of the communities where less than 50% of nodes belonged to the same category, it was discovered that the two most common categories within each community were broadly similar. This suggests that while there may be greater diversity in the types of nodes present in these communities, there are still underlying similarities that link them together. Identifying these similarities can provide valuable insights into the overall structure and dynamics of the dark web ecosystem, as well as inform strategies for addressing potential risks or threats associated with different categories of URLs.

Figure 4 is a stacked bar chart that shows the top two categories that make up the majority of nodes within each community. It is interesting to note that in many cases, the top two categories are similar or related. For example, the "Business" and "E-commerce" categories often appear together, suggesting that these types of nodes are closely linked within the dark web ecosystem. Similarly, the "Computers and Technology" category appears frequently alongside "Social Networking and Messaging," which may indicate that these two categories share common characteristics or serve similar purposes within the dark web.

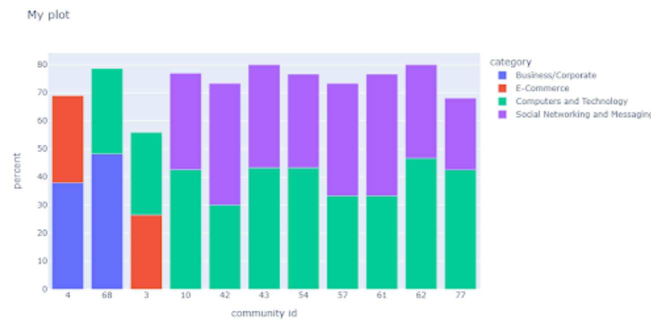

Fig. 4. Stacked Bar Chart Showing Top Two Categories in Each Louvain Community in Original Dataset

b) On Subset Dataset: A total of 2722 nodes formed 128 communities in the subset dataset, with 123 communities having over 50% of nodes belonging to the same category, after applying the Louvain algorithm. The community structure formed by the Louvain algorithm on the subset dataset is depicted in Figure 5.

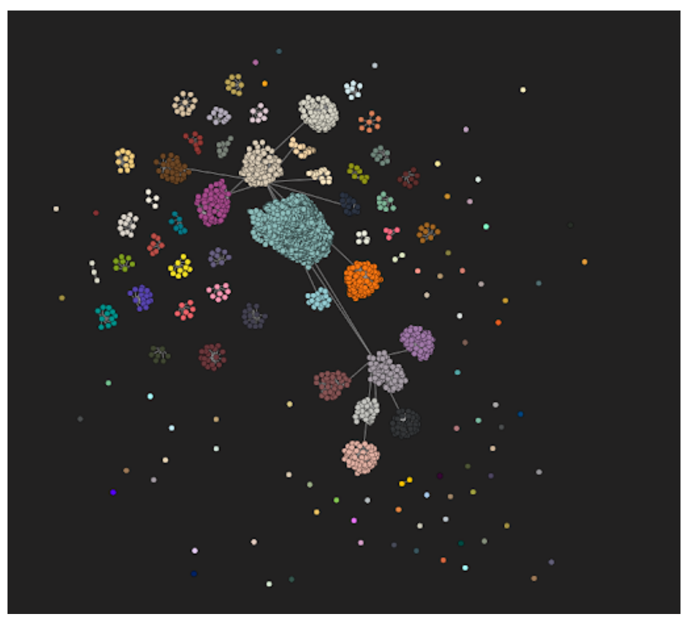

Fig. 5. Community created by Louvain Algorithm on Subset Dataset

Figure 6 displays the percentage of nodes belonging to the same category in the communities.

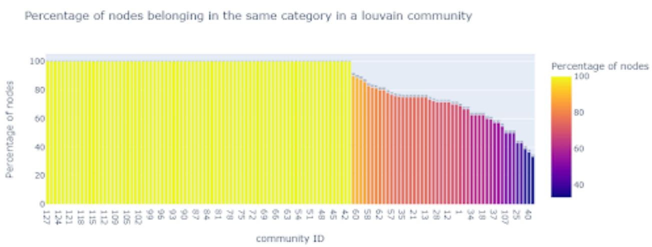

Fig. 6. Bar Chart of percent of nodes belonging to same category in a Louvain community in the Subset Dataset

Algorithm 2 Leiden Community Detection Algorithm

Require: A network represented as a graph $G = (V, E)$, where V is the set of nodes and E is the set of edges.
Ensure: A community assignment for each node in the network.

```
Initialize the quality function Q to be the modularity of the initial partition of nodes.
Initialize the partition of nodes randomly or based on some prior knowledge.
Set the flag improved to be true.
while improved is true do
    Set improved to be false.
    for each node i ∈ V do
        Calculate the modularity gain ΔQ of moving node i to each of its neighboring communities.
        Move node i to the community with the highest modularity gain ΔQ.
        if the modularity of the partition has improved then
            Set improved to be true.
        end if
    end for
end while
Output the final community assignment for each node.
```

2) *Leiden Algorithm:* The Leiden algorithm is similar to the Louvain algorithm in that it also optimizes modularity by iteratively merging communities. However, the Leiden algorithm employs a refinement step in which communities are iteratively refined to improve modularity. At each step, the algorithm evaluates the modularity gain of moving each node to its neighboring communities, and the node is moved to the community with the highest modularity gain. This process continues until no further improvement in modularity is possible. The Leiden algorithm can also be applied to the network of dark web URLs to detect communities based on the similarity of URLs.

a) On Original Dataset: Figure 7 displays the communities generated after applying the Leiden algorithm to the original dataset. Each node in the graph represents a dark web URL, and the communities are formed by grouping nodes that have similar characteristics or functionalities. The Leiden algorithm successfully identifies multiple communities that are well-defined and distinct from each other. The community structure is evident from the clear separation of groups in the graph.

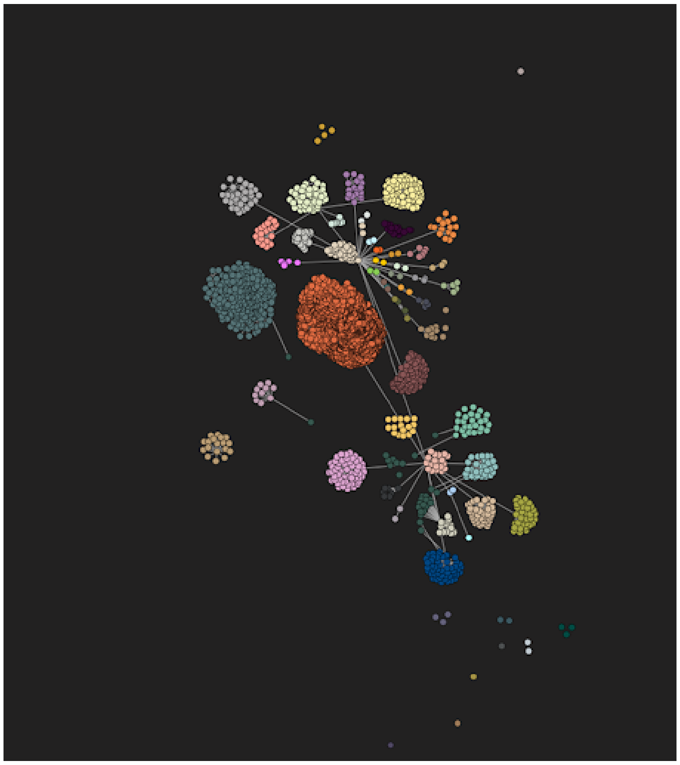

Fig. 7. Community created by Leiden Algorithm on Original Dataset

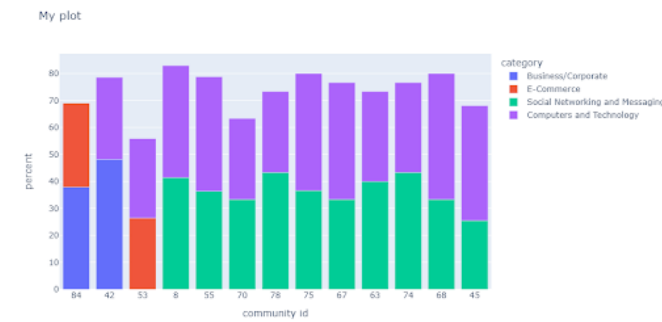

Fig. 9. Stacked Bar Chart Showing Top Two Categories in Each Leiden Community in Original Dataset

b) On Subset Dataset: Using the Leiden algorithm on the subset dataset, a total of 2722 nodes formed 128 communities, with 123 communities having more than 50 percent of the nodes belonging to the same category. Figure 10 represents the communities created in a graphical form by the Leiden algorithm on the subset dataset and Figure 11 displays the percentage of nodes belonging to the same category in the communities.

Figure 8 represents the bar chart for the percent of nodes using the Leiden algorithm on the subset dataset. Upon further examination of the communities where less than 50% of nodes belonged to the same category, it was discovered that the two most common categories within each community were broadly similar. Figure 9 is a stacked bar chart that shows the top two categories that make up the majority of nodes within each community.

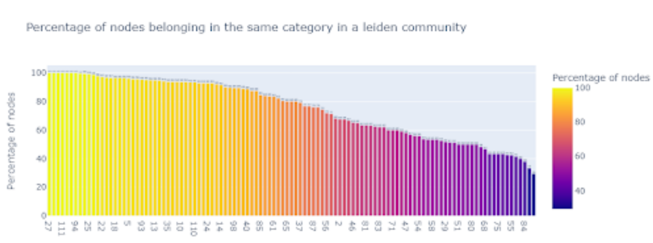

Fig. 8. Bar Chart of percent of nodes belonging to same category in a Leiden community in Original Dataset

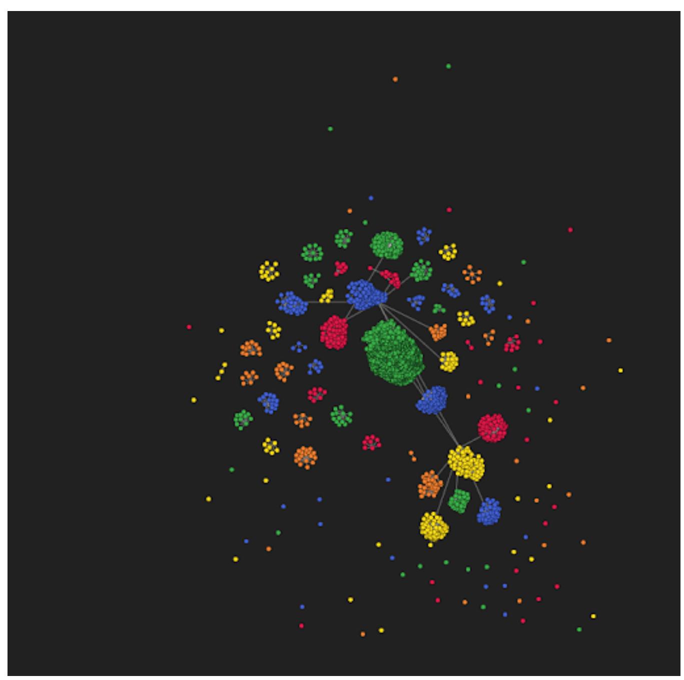

Fig. 10. Community created by Leiden Algorithm on Subset Dataset

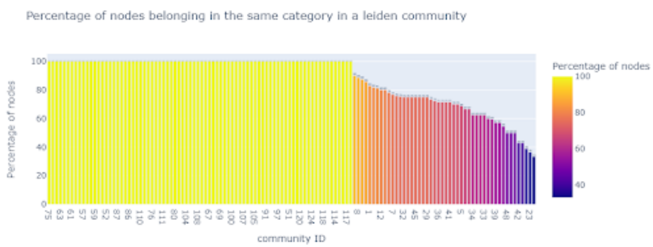

Fig. 11. Bar Chart of percent of nodes belonging to same category in a Leiden community in Subset Dataset

Comparison of Louain and Leiden Algorithms: Modularity is a measure of the quality of a network partition into communities or modules. It evaluates the degree to which nodes within the same community are more densely connected to each other than to nodes in other communities. The modularity score ranges from -1 to 1, with higher values indicating a better community structure.

In the context of community detection algorithms, coverage refers to the proportion of nodes in the network that are assigned to at least one community. A higher coverage value indicates that more nodes are assigned to communities, which can suggest that the algorithm is detecting more meaningful community structure in the network.

The conductance is a ratio between the graph's cut size and its volume, hence its values are always between 0 and 1. A conductance of 0 implies that there are no edges between nodes in different communities, whereas a conductance of 1 indicates that there are no connections within the same community. A conductance below 0.5 typically indicates a healthy community structure, but it may vary by application.

Table 1 displays the modularity, coverage, and conductance scores obtained by the Leiden and Louvain algorithms on the two datasets (Original and Subset). The results of the Louvain and Leiden algorithms on both the original and subset datasets were compared in Table 1. In the original dataset, both algorithms achieved high modularity and coverage scores, with Leiden having a slightly higher conductance score. In the subset dataset, the modularity and coverage scores were lower for both algorithms, indicating that the community structure is not as strong as in the original dataset but Leiden had a slightly higher coverage score compared to Louvain. The conductance score was very low for both algorithms in the subset dataset, indicating a strong community structure. Overall, the results indicate that both algorithms perform well in detecting communities in the dark web datasets, with Leiden having a slight edge over Louvain in terms of conductance and coverage scores.

TABLE I
COMPARISON OF LOUVAIN AND LEIDEN ALGORITHMS

Algorithm	Dataset	Modularity	Coverage	Conductance
Louvain	Original	0.941	0.987	0.01
Leiden	Original	0.942	0.987	0.011
Louvain	Subset	0.58	0.99	0.005
Leiden	Subset	0.581	0.995	0.005

3) *Label Propagation Algorithm:* In the context of the dark web URLs dataset with 16 categories and its subset dataset with 5 categories, the Label Propagation Algorithm can be applied to detect communities of URLs based on their similarities in terms of categories. Using label propagation to predict the categories of unlabeled nodes is a common technique in semi-supervised learning. The idea is to use the labeled examples to train a model or algorithm, and then use this model to make predictions on the unlabeled examples. The algorithm starts with assigning a unique label to each node representing a URL and then iteratively propagates the labels to its neighboring nodes based on the most frequent label in the neighborhood. This process continues until convergence or a maximum number of iterations is reached. The algorithm can provide insight into the patterns and structures of URLs within each category and can help identify hidden relationships between different categories of URLs.

Algorithm 3 Label Propagation Algorithm

Require: A network represented as a graph $G = (V, E)$, where V is the set of nodes and E is the set of edges. Each node in V is assigned a unique label and belongs to one of the predefined categories.

Ensure: A community assignment for each node in the network based on the most frequent label in its neighborhood.

```
Initialize each node i in V with a unique label and its category.
repeat
    for each node i ∈ V do
        Let S_i be the set of i's neighbors in G.
        Let L_j be the label of node j ∈ S_i.
        Count the frequency of each label in S_i.
        Set the label of node i to the most frequent label in S_i.
    end for
until convergence or maximum number of iterations is reached
Output the community assignment for each node based on its label.
```

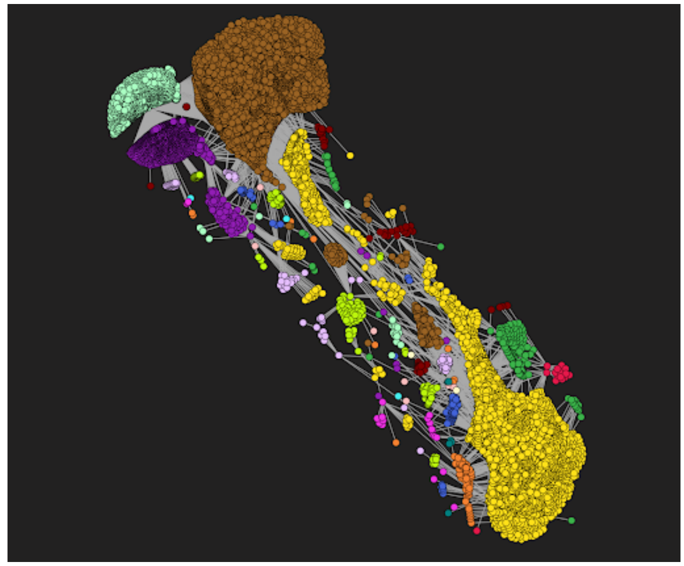

Fig. 12. Community created by Label Propagation Algorithm on Original Dataset

a) On Original Dataset: In this case, label propagation is used to predict the categories of 3000 unlabeled nodes based on the categories of their neighbors in the network of 9386 nodes. This is a reasonable approach, as label propagation has been shown to perform well in semi-supervised settings, particularly when the network is sparse or when the labels are correlated with the network structure. The resulting community graph generated by label propagation on the original dataset can be observed in Figure 12.

The overall accuracy of the model was found to be 0.68, which indicates that the model is able to predict the correct category for 68% of the nodes. Label Propagation algorithm demonstrated reasonable performance in predicting categories of unlabeled nodes. The results provide valuable insights into the structure and organization of the dark web. These findings could have important implications for law enforcement and cybersecurity professionals working to monitor and mitigate illicit activities on the dark web. The associated confusion matrix is depicted in Figure 13.

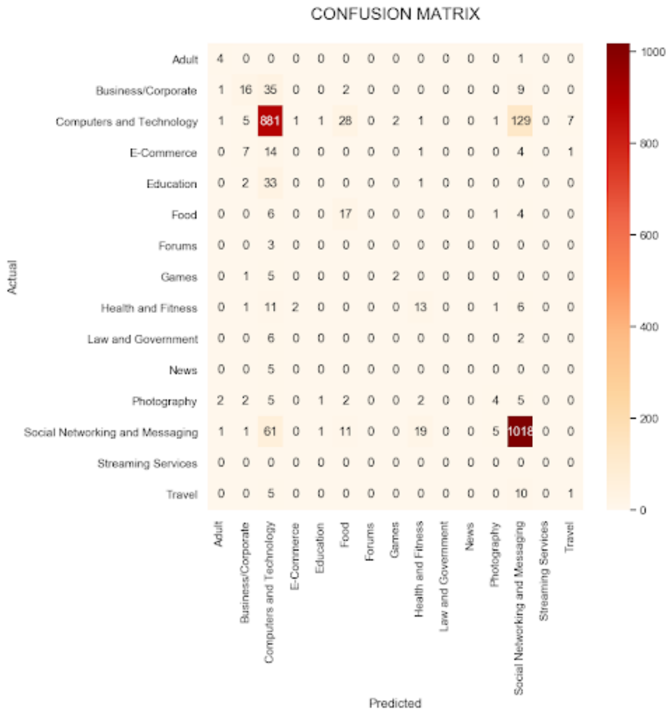

Fig. 13. Confusion Matrix for Label Propagation Algorithm on Subset Dataset

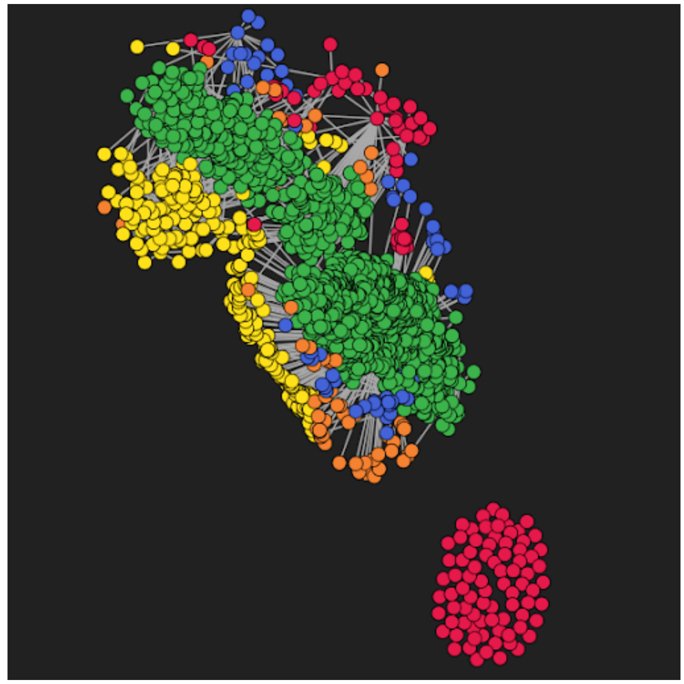

Fig. 14. Community created by Label Propagation Algorithm on Subset Dataset

b) On Subset Dataset: In this case, label propagation is used to predict the categories of 875 unlabeled nodes based on the categories of their neighbors in the network of 2772 nodes. This is a reasonable approach, as label propagation has been shown to perform well in semi-supervised settings, particularly when the network is sparse or when the labels are correlated with the network structure. The resulting communities generated by the Label Propagation algorithm on the subset dataset can be observed in Figure 14.

The accuracy of a classification algorithm is a measure of how well it is able to predict the correct class or category for the input data. In the case of label propagation algorithm applied to the subset dataset, the overall accuracy was found to be 0.76. This means that the algorithm was able to correctly predict the category of 76% of the nodes in the dataset. Figure 15 depicts the confusion matrix for label propagation algorithm applied on the subset dataset.

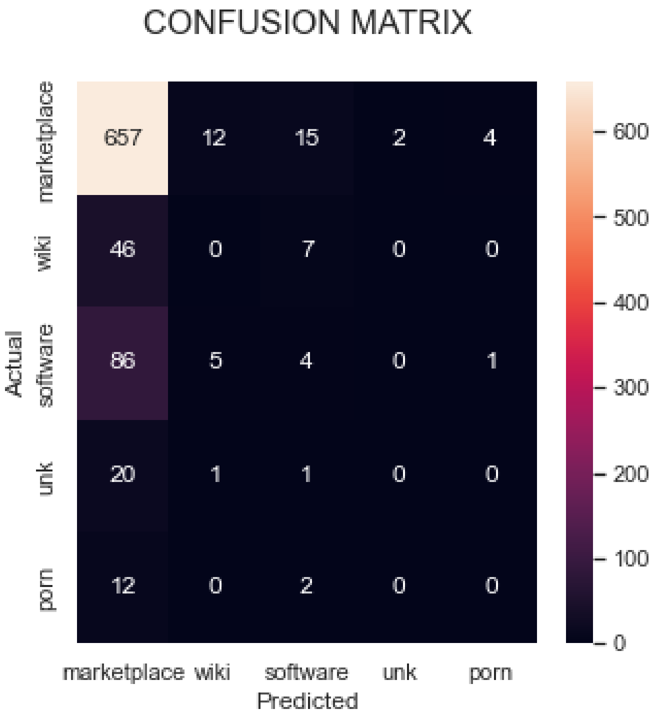

Fig. 15. Confusion Matrix for Label Propagation Algorithm on Subset Dataset

IV. Conclusion

In conclusion, this research paper presents a comprehensive dataset of textual content from various tags on dark web pages, and utilizes graph-based community detection algorithms to analyze the structure and organization of the dark web. The results showed that both Leiden and Louvain algorithms performed similarly well in detecting communities in the original dataset with high modularity and coverage scores. However, when applied to the subset dataset, the modularity and coverage scores decreased significantly. On the other hand, the Label Propagation algorithm demonstrated reasonable performance in predicting the categories of unlabeled nodes, achieving an accuracy of 68% and 76% on the original and subset datasets, respectively. The findings provide valuable insights into the organization and categorization of illicit activities on the dark web, which could have important implications for law enforcement and cybersecurity professionals working to monitor and mitigate such activities. Overall, this research contributes to a better understanding of the dark web and its underlying network structure.

References

[1] Soro, Francesca, et al. *"Sensing the noise: Uncovering communities in darknet traffic."*, 2020 Mediterranean Communication and Computer Networking Conference (MedComNet). IEEE, 2020.

[2] Ríos, Sebastián A., and Ricardo Muñoz. *"Dark web portal overlapping community detection based on topic models."*, Proceedings of the ACM SIGKDD workshop on intelligence and security informatics. 2012.

[3] L'huillier, G., Alvarez, H., Ríos, S.A. and Aguilera, F., 2011. *Topic-based social network analysis for virtual communities of interests in the dark web.*, ACM SIGKDD Explorations Newsletter, 12(2), pp.66-73.

[4] Que, X., Checconi, F., Petrini, F. and Gunnels, J.A., 2015, May. *Scalable community detection with the louvain algorithm.* In 2015 IEEE International Parallel and Distributed Processing Symposium (pp. 28-37). IEEE.

[5] Anuar, S.H.H., Abas, Z.A., Yunos, N.M., Zaki, N.H.M., Hashim, N.A., Mokhtar, M.F., Asmai, S.A., Abidin, Z.Z. and Nizam, A.F., 2021, December. *Comparison between Louvain and leiden algorithm for network structure: A review.*, In Journal of Physics: Conference Series (Vol. 2129, No. 1, p. 012028). IOP Publishing.

[6] Garza, Sara E., and Satu Elisa Schaeffer. *"Community detection with the label propagation algorithm: a survey."* , Physica A: Statistical Mechanics and its Applications 534 (2019): 122058.

[7] Gregory, S., 2010. *Finding overlapping communities in networks by label propagation.* New journal of Physics, 12(10), p.103018.