

Illicit Darkweb Classification via Natural-language Processing: Classifying Illicit Content of Webpages based on Textual Information

Giuseppe Cascavilla¹, Gemma Catolino² and Mirella Sangiovanni²

¹*Eindhoven University of Technology, Jheronimus Academy of Data Science, The Netherlands*

²*Tilburg University, Jheronimus Academy of Data Science, The Netherlands*

Keywords: Natural-language Processing, DarkWeb, Bert, RoBERTa, Machine Learning, ULMFit, LSTM, AI.

Abstract: This work aims at expanding previous works done in the context of illegal activities classification, performing three different steps. First, we created a heterogeneous dataset of 113995 onion sites and dark marketplaces. Then, we compared pre-trained transferable models, i.e., ULMFit (Universal Language Model Fine-tuning), Bert (Bidirectional Encoder Representations from Transformers), and RoBERTa (Robustly optimized BERT approach) with a traditional text classification approach like LSTM (Long short-term memory) neural networks. Finally, we developed two illegal activities classification approaches, one for illicit content on the Dark Web and one for identifying the specific types of drugs. Results show that Bert obtained the best approach, classifying the dark web's general content and the types of Drugs with 96.08% and 91.98% of accuracy.

1 INTRODUCTION

The Internet, in recent times, dominates everyone's daily and professional lives, and it can be divided into three parts: The surface Web, the Deep web, and the Dark web. The surface web is the well-known part of the Internet that most of us use every day. The Deep web is unavailable is hidden from commercial search engines, e.g., Google, since its content cannot be indexed by web crawlers. It is common to believe that the deep and the dark web belong to the same concept; however, the Dark web (Cascavilla et al., 2021) is publicly available but can only be accessed with an encryption tool, e.g., Onion Router (Tor) (Mansfield-Devine, 2014). Tor provides "hidden services" in order to host websites anonymously.

This anonymity creates the perfect environment where illegal endeavors can occur. This renders the dark web investigation extremely enticing to both law enforcement agencies and researchers to support them. Indeed the literature on this topic ranges from user identification (Spitters et al., 2015), criminal motivation (Dalins et al., 2018), and content analysis of Tor (Biryukov et al., 2014) to product categorization for Darknet marketplaces (Graczyk and Kinningham, 2015) using natural languages processing techniques (NLP), e.g., Bert. Nonetheless, most of the previous works (Graczyk and Kinningham, 2015; Spitters et al., 2015) dealt with activities from one marketplace and did provide an extensive evaluation in terms

of accuracy when comparing different text classification approaches. This study aims to expand previous studies in three different steps. First, we expanded previous datasets provided in the past, expanding their size with new instances from different Darknet marketplaces. Then we provided a deeper comparison of pre-trained transferable models, e.g., ULMFit (Universal Language Model Fine-tuning), with a traditional text classification approach like LSTM (Long short-term memory) neural networks. Finally, we built two approaches for classifying the Dark Web's illicit activities and types of drugs.

The results show how the Bert model outperformed ULMFit and LSTM during the testing phase for both the models, i.e., illicit activities and drugs classes, while RoBERTa model obtained the lowest accuracy. According to our results, our models achieve 96.08% accuracy for classifying illegal activities from more than one marketplace.

2 RELATED WORKS

Text represents the main feature for analyzing and classifying the Darknets, thus implying the usage of text mining and Natural Language Processing techniques. For instance, Latent Dirichlet Allocation (LDA) (Blei et al., 2003) has been widely applied by the research community for identifying (i) forum

discussions (Tavabi et al., 2019), or (ii) topics from Darknet (Yang et al., 2009). Additional studies used Bag of Words (BOW) (Tsai, 2012) and Term Frequency Inverse Document Frequency (TF-IDF) (Yuntao et al., 2005) for categorizing the content of the Darknet. For example, Nabki et al. in (Al Nabki et al., 2017) created a dataset that encompasses the activities on the Darknet web pages. They found that combining TF-IDF words representation with the Logistic Regression classifier achieves 96.6% accuracy. On the same line, Graczyk et al. in (Graczyk and Kinningham, 2015) attempt to provide illicit product categorization within a marketplace, using a machine learning approach. Choosing TF-IDF for feature extraction, the model achieved 79% accuracy. Besides these positive results, Choi et al. (Choi et al., 2014) showed how statistical methods like TF-IDF fall behind in text classification because they cannot comprehend the semantic meanings of the text created by people. For this reason, Sabbah et al. (Sabbah et al., 2015) introduced a Hybridized term-weighting method to accurately identify terrorism activities with textual content from the Dark web.

3 METHODOLOGY

The following subsections describe the methodology of our study. We report all the steps done and the technology used. All the experiments and data manipulation has been conducted within the Python programming environment.

Goal and Contribution. Our study aims to improve the state of the art when classifying illicit activities on the Dark web pages based on textual information. In particular, we expanded the previous datasets provided by past studies (Al Nabki et al., 2019; Graczyk and Kinningham, 2015). Then, we provide a deeper comparison of three pre-trained transferable models, e.g., ULMFit, with a traditional text classification approach like LSTM neural networks. Finally, we construct two machine learning approaches to classify the Dark Web’s illicit activities and the types of drugs and analyze the performance.

3.1 Dataset Creation

Data Collection. This paper uses data from various sources from the Dark web and some from the surface web. The dataset is built to combine existing datasets (Agora, Duta10k (Al Nabki et al., 2019)) and crawl the web specifically for this study. In more detail, the different data collection procedures can be divided

into four parts.

The Duta10k dataset is an extension of the Duta one (Al Nabki et al., 2017). Compared to the Duta dataset, Agora did not contain onion addresses. Hence, we considered only titles and descriptions. Nonetheless, training a text classifier on the product description is challenging since the model should be able to get textual information focused only on titles and the descriptions of the product page.

In the context of our study, we boosted this dataset manually, extracting further pages. In particular, we collected manually several onion sites by surfing the Tor browser. As a result, we downloaded 148 onion sites. Their content ranged from counterfeit personal identification accounts (e.g., PayPal accounts), cards, and drugs and violence (weapons, hitman services). The downloaded onion pages belong to a specific onion site that cataloged all the available sites in the Tor network distributed per topic. After collecting these sites, we relied on an automatic solution for collecting HTML pages containing illicit activities from the Surface Web. The tool used to automate the collection procedure is HTTrack tool¹. The copied website consists of images, links, code, and HTML pages (Engebretson, 2013). We downloaded web sites from the Surface Web like *lifakes.com* and *buypurecocaineonline.com*. Finally, we merge other data crawled by us from the Dark web, i.e., Berlusconi and the Silkroad markets.

After creating the dataset, we proceeded with the labeling phase and the data fusion process.

- Duta10k: the labeling process was already provided by the authors from (Al Nabki et al., 2019). The labels have been modified in order to improve the categories;
- Agora: we used the *product descriptions* and *titles* of the HTML pages to create new labels or to fill the labels provided by Duta10k.
- Other data sources: we manually scraped them. After extracting the text, they are hand-labeled using previous labels or creating new labels.

The Duta10k dataset initially had 26 main labels that the authors manually assigned in (Al Nabki et al., 2019). Some of them were also assigned a sub-label. A detailed description of it can be found in table 5 in the online appendix (Appendix, 2022), specifically in the first two columns. We kept most of the original labels from the Duta10k as main classes, e.g., Services, Counterfeit Personal Identification, Counterfeit Credit-Cards, Counterfeit Money, Leaked Data, Porno, Drugs, and Violence (not included in the table for the sake of space). One significant modifica-

¹<https://www.httrack.com>

tion concerned the Social Network class. In (Al Nabki et al., 2019), the authors appointed four sub-labels to it, namely, Chat, Blog, Email, and News. The Blog was altered to the main label, and the others were removed. We ignored Marketplace label due to very few occurrences.

After carefully studying each label, we merged the Hacking class with the Services one because the former referred mainly to providing hacking services. The Forum’s main class was transferred from the new main classes to the Social Network main class because forums were considered a form of communication like the social networks. Therefore, having an individual class for it seemed redundant. Similarly, we added the Human Trafficking main initial main class to the new Violence main class. Furthermore, we ignored the Cryptocurrency class as its content varied from site to site and the authors’ reasons for creating this label were unclear. Lastly, we eliminated the Cryptolocker class. All these changes to the original labels were made after considerable examination of the content of their corresponding sites. At the end of this process, we created a list with the changed labels, and we mapped them to the initial ones. Labels like Info, eBooks, Relationships and Sex, and Drug paraphernalia and Pipes are transformed into Library Information and Drugs Paraphernalia main classes.

Agora Category Labels Modifications. The Agora web page categorization was more detailed than the Duta10k dataset. To adapt these categories to our work, we applied several modifications. For the sake of space, we provide a short example of all the changes made with the labels from the Agora dataset. After carefully investigating each category and its content, all the changes were made and compared to the Duta10k main labels. We renamed the *Service/Hacking* category from the Agora as *Services*. Same for the category *Drugs/Psychedelics/2C* that became *Drugs*. *Info/eBooks/Philosophy* was renamed as *Library Information*. For a more detailed overview please refer to table 6 in the online appendix (Appendix, 2022). After being transformed, the labels from Agora decreased from 104 to 15. We introduced three new labels from Agora to the final classes that did not exist before in the Duta10k, namely, Counterfeit Products, Counterfeit Coupons, and Accounts. We renamed the Agora labels to normalize the dataset with more details to the Duta10k dataset and introduce new categories.

Drugs Labels. During the labeling procedure from both Agora and Duta10k, we discovered several

sub-labels describing in more detail the main labels. However, we considered only sub-classes referring to drugs for the classification task. There are two different explanations of why only drug sub-classes were considered. Firstly, using all the sub-labels implied that the classification methods should have been modified to a hierarchical format. Secondly, according to the latest studies, the interest in drug sales increases on the Dark Web. In particular, researchers (Bhaskar et al., 2017), (Décary-Héту et al., 2018), (Celestini et al., 2016) and (Minnaar, 2017) investigated marketplaces like Silkroad for their illicit drugs content. We created drug sub-classes from the detailed descriptions of the Agora labels. The finalized drugs sub-classes are 49, and a detailed depiction of them can be found in table 7 in the online appendix (Appendix, 2022). All these sub-classes correspond only to the Drugs main class.

Final Classes. As previously mentioned, the final labels are generated from blending and normalizing the Agora with the Duta10k dataset. The other data sources, particularly the self-collected pages, were manually labeled after creating the final classes. Table 1 presents the final 19 main classes.

Table 1: The 19 Main Classes of the Final Dataset.

Final Main Classes	
Accounts	Drugs
Counterfeit Coupons	Drugs paraphernalia
Counterfeit	Fraud
Credit-Cards	
Counterfeit Money	Leaked Data
Counterfeit Other	Library Information
Counterfeit	
Personal-Identification	Porno
Counterfeit Products	Services
Cryptocurrency	Services/Money
Violence	Social Network
Substances for Drugs	

Final Dataset. The complete dataset contains textual information from the Duta10k dataset, the Agora, and the manually collected pages. The pages manually collected contain 640 HTML pages from Canna-Home, 323 from Berlusconi market, and 1660 from Silkroad, plus other 149 manually collected pages from the Dark Web (we used only 50 pages due to the relationship to our labels). From the pages collected using the HTTrack tool, 120 are incorporated inside the dataset from the Normal Web. The rest of the text is derived from the Agora dataset. Table 2 illustrates the final dataset formation. Figure 1 depicts the distribution of the classes for the whole dataset. In (Appendix, 2022), figure 4 shows the distribution without the Drugs class.

Table 2: The Data Sources in the Final Dataset.

Data-source	Data Instances
Agora	108261
Berlouskoni	323
CannaHome	640
Duta10k	2941
Manual Dark Web Collection	50
Normal Web	120
SilkRoad	1660
Total of Instances Used	113995

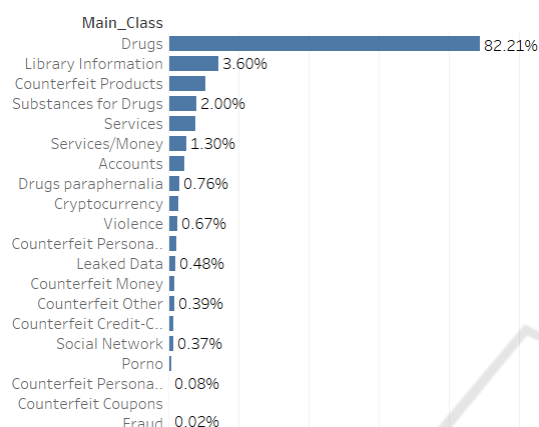


Figure 1: The Distribution of the 19 Main Classes for the Final Dataset.

3.2 Text Preparation and Extraction

This subsection presents all the steps followed to prepare the text for the language classification model. First, we extracted the text from the HTML pages to feed these data to the language model. We used BeautifulSoup² library to extract the text in Python. BeautifulSoup is a Python library used to survey and extract HTML content easily (Hajba, 2018). Also, even though all the HTML pages are stored locally, we used the module-urllib.request³ Python library to open the URLs. Then we built two different approaches based on the market, one for random pages from the Dark web and one specific for marketplaces such as Silkroad and Berlusconi.

In the first approach to reading the HTML files, we called the urllib.request Python library. Next, we used BeautifulSoup Python library to perform the following steps:

1. Eliminate all the script and style elements.
2. Get the rest of the text from the HTML.
3. Break the text into lines and remove the leading and trailing space on each of them.
4. Break each of the multi-headlines into a line.

²<https://www.crummy.com/software/BeautifulSoup/>

³<https://docs.python.org/3/library/urllib.request.html>

5. Drop any blank or empty lines.

The second approach intends to extract only the description and the title from the HTML pages and refers only to specific marketplaces. The text from HTML from marketplaces contains repeated words (e.g., the marketplace name) and, in general, has some fixed layout. We eliminated repeated words by focusing only on the description and title. We also removed the pages that only include listings of products and do not describe a specific item. This technique was not used for all pages regardless if they belong to a marketplace because using it requires inspecting the pages, and it would not generalize well. It was utilized for marketplaces only because they have a repeated template for all the pages. Conversely, the random pages from the Dark web have a different format. Finally, after the text derived from both approaches was ready, we moved to the preparation's next part.

Text Pre-processing. Preprocessing textual data is a key component of any text classification task. The preprocessing part is usually comprised of tokenization, stop-word removal, lowercase conversion, and stemming (Uysal and Gunal, 2014). Nonetheless, the tokenization step differed from the NLP techniques used in this work. The preprocessing procedure followed in this work can be divided into six steps.

1. Remove any HTML tags using BeautifulSoup Python library.
2. Remove URLs using the *re* library that provides regular expression matching operations.
3. Converting contractions; i.e., when *you're* becomes *you are*.
4. Remove all special characters like currency signs (e.g., \$) and words that include numbers.
5. Preprocessing, to remove articles, prepositions, and pro-nouns (Vijayarani et al., 2015). Often these are not required for tasks such as sentiment analysis or text classification. Consequently, we removed stopwords using the collection from the NLTK (Bird et al., 2009) Python library.
6. In the final step, we used the WordNetLemmatizer from the NLTK library to lemmatize the processed text. Lemmatization is a way to normalize the dataset, i.e., from *bought* to *buy*. Lastly, all the text is transformed to lowercase.

3.3 Research Model

For each of the techniques, we constructed two different models, i.e., one for predicting illicit activities

and one for the type of drugs. Due to the space limitation, we just reported them without details. In particular, we used Long Short Term Memory, Universal Language Model Fine-tuning, Bidirectional Encoder Representations, and Robustly optimized BERT approach. We evaluated the performance of the models, i.e., classifying illegal activities and types of drugs, through (i) accuracy and the confusion matrix, (ii) precision and recall of each model, and (iii) training time and computing power. The transfer learning models are supposed to work with few data instances. Therefore, all four models are tested in the whole dataset (meaning the Main Classes) and a subset of the dataset (the Drugs Classes) to examine this attribute. Lastly, the procedure follows the same steps as for the classification of the types of drugs.

4 RESULTS

In this section, we report the results achieved. We discuss the performance of the models during training and the comparison between the methods regarding the testing process, for each technique used for each model, i.e., main classifier (illicit activities) and drug types. Due to the space limitation, we report only plots of the best models, i.e., BERT. The rest is available in the online appendix (Appendix, 2022).

LSTM. Both models are compiled with the same model configurations meaning the loss function, optimizer, and metrics. Specifically, we chose categorical cross-entropy for the loss function, Adam as an optimizer, and accuracy as a reference metric. The models are trained for five epochs with a batch size equal to 32. Nonetheless, the results from the training process of the Main Classes LSTM model and the Drugs Classes LSTM are notably different. First of all, the computing time varies per model, as the Main Classes model needed approximately 1 hour and 24 minutes to train while the Drugs classifier took less than one hour.

ULMFit. Starting with the first cycle, the last layer of the pre-trained ULMFit is unfrozen, and the model is trained for two epochs with a learning rate equal to $1e - 01$. During the first batches of data, the validation loss is much higher than the training loss; so the model struggles to learn the validation data. In the second cycle, the last two layers are unfrozen; indeed there are parts of the data batches that seem easier for the model to learn. Lastly, in the third cycle, all the layers are used to train the model for five epochs, and it learns better than before. The

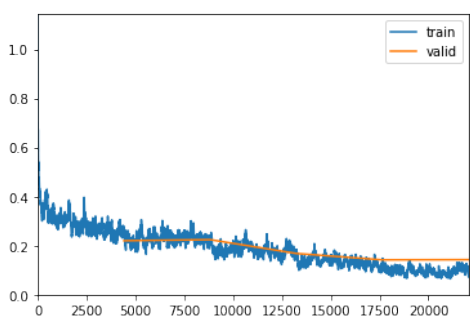
training time took half hour. Like the Main Classes model, the Drugs Classes model is trained during three cycles, where the last layer, the two last layers, and the whole architecture are unfrozen, respectively. We used the same epochs per cycle as ULMFit. Both ULMFit classifiers overfit and have greater loss values than their LSTM counterparts.

BERT. The Main Classes Bert model is trained for two cycles. In the first cycle, all the layers of the architecture are frozen except for the last two. Next, the model is trained for five epochs with a $2e-4$ maximum learning rate. We can notice that the model is training adequately. There is no indication of overfitting, and the loss values are below 0.4. In the second and last cycle, the last three layers are unfrozen, and it is trained for three epochs with a lower maximum learning rate, namely $2e-5$. In this cycle, the classifier starts to overfit on the training data batches probably because of the layers used, and the model becomes complex. This model took approximately one and a half hours to train. Finally, Bert is trained with the same configurations as the Main Classes one in the Drugs Classes. However, in the second cycle with three unfrozen layers, this Bert model presents less overfitting than the previous one but higher loss. A possible reason is that the Drugs Classes dataset is smaller than the whole dataset used in the other Bert. Consequently, fewer data might improve the training results. Drugs Classes took less than an hour. So far, both Bert approaches outperform the ULMFit and LSTM models in training. The loss values are low, and the training and validation data batches are more appropriately learned.

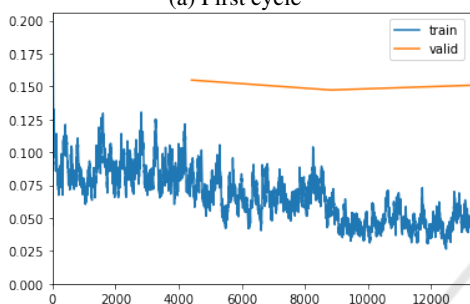
RoBERTa. The Main Classes RoBERTa model is trained for three cycles; the model overfits on the third and last cycle instead of Bert that overfits significantly from the second cycle. The configurations for the first two cycles are the same as for the Bert models. In the third cycle, all the layers are unfrozen, and the model ends up learning better the training batches than the validation batches. RoBERTa model in the second cycle is more prone to overfitting than the corresponding Bert. To conclude, we can indicate that the RoBERTa and Bert models for the Main and Drugs classes present superior results than the ULMFit and the LSTM.

Classification Methods Performance Comparison.

The overall performances of the Main Classes are shown in table 3, while the ones of the Drugs in table 4. The tables present evaluation metrics, particularly accuracy, precision, recall, and f1 score.

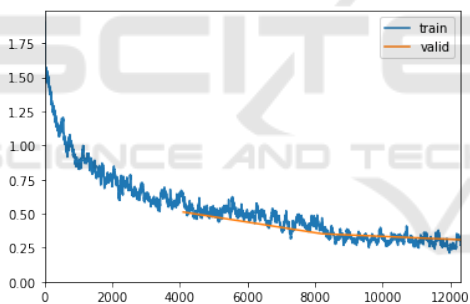


(a) First cycle

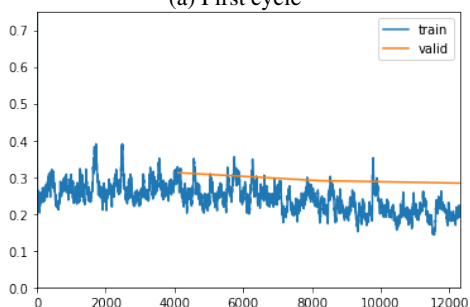


(b) Second cycle

Figure 2: Main Classes Bert model per learning cycle.



(a) First cycle



(b) Second cycle

Figure 3: Drugs Classes Bert model per learning cycle.

Moving to the Drugs Classes models results in table 4, Bert has the highest accuracy score. However, the metrics scores for these models are more diverse than the previously discussed. Specifically, the ULM-

Table 3: Results metrics for the Main Classes Models.

Main-Classes Models	Accuracy	Precision	Recall	F1 score
Bert	96.08%	0.82	0.78	0.80
RoBERTa	95.78%	0.85	0.82	0.84
ULMFit	95.98%	0.77	0.74	0.74
LSTM	95.91%	0.80	0.77	0.78

Table 4: Results metrics for the Drugs Classes Models.

Drugs-Classes Models	Accuracy	Precision	Recall	F1 score
Bert-Drugs	91.98%	0.85	0.85	0.84
RoBERTa-Drugs	91.38	0.83	0.79	0.80
ULMFit-Drugs	61.01	0.23	0.19	0.18
LSTM-Drugs	90.58	0.84	0.82	0.82

Fit model severely underperforms compared to the rest. The Bert surpasses all three models in the metrics scores, but more importantly, the precision and recall values are the same. This is an ideal trade-off situation as the precision and recall metrics should be as close as possible. Bert is the best from the Drugs Classes models, and RoBERTa comes second. The results of the Drugs Classes models can be used to evaluate their scalability. The scalability of the models refers to how well they perform with fewer data instances. Since the Drugs Classes ULMFit model performs poorly compared to the Main Classes one, it is clear that the ULMFit approach is not scalable. The other three models have produced similarly adequate results with the whole dataset and a subset of the datasets, specifically, the drugs-related HTML pages.

Based on the results, we investigated the confusion matrix of Bert and RoBERTa approaches for both models. Unfortunately, due to the space limitation, we did not add figures, but we uploaded them in the online appendix (Appendix, 2022). When investigating the Main Classes Bert matrix, we noticed that the model mostly confuses the label Drugs with the label Substances for Drugs. In particular, when the label should have been Substances for Drugs, Bert assigns the label Drugs 26 times and vice versa. Regarding the RoBERTa model, the confusion matrix indicates that the Drugs and Substances for Drugs labels are the most confused, as in Bert. However, in RoBERTa, the Drugs and Substances for Drugs classes are wrongly identified 67 times. Therefore, the ability of Bert to identify the correct labels seems better.

5 CONCLUSION

This study provided several different approaches for classifying the Dark Web content that achieved good results. Specifically, the best model classified the Dark Web’s general content with 96.08% accuracy and the specific types of drugs with 91.98% accuracy. Future studies include exploiting the hierarchi-

cal classification model and developing a similar classifier for multiple languages.

ACKNOWLEDGEMENTS

The authors thank MSc student Theodora Tzagkaraki for her valuable job, and Prof. W.J. van den Heuvel and Prof. D.A. Tamburri for providing feedback to improve the quality of the paper.

REFERENCES

- Al Nabki, W., Fidalgo, E., Alegre, E., and Fernández-Robles, L. (2019). Torank: Identifying the most influential suspicious domains in the tor network. *Expert Systems with Applications*, 123.
- Al Nabki, W., Fidalgo, E., Alegre, E., and Paz, I. (2017). Classifying illegal activities on tor network based on web textual contents. pages 35–43.
- Appendix (2022). Illicit darkweb classification via natural-language processing: Classifying illicit content of webpages based on textual information <https://figshare.com/s/54a17898301e2c9f7ca9>.
- Bhaskar, V., Linacre, R., and Machin, S. (2017). The economic functioning of online drugs markets. *Journal of Economic Behavior & Organization*.
- Bird, S., Klein, E., and Loper, E. (2009). *Natural language processing with Python: analyzing text with the natural language toolkit*.
- Biryukov, A., Pustogarov, I., Thill, F., and Weinmann, R. (2014). Content and popularity analysis of tor hidden services. In *IEEE 34th International Conference on Distributed Computing Systems Workshops (ICDCSW)*, pages 188–193.
- Blei, D. M., Ng, A. Y., and Jordan, M. I. (2003). Latent dirichlet allocation. *Journal of machine Learning research*, 3(Jan):993–1022.
- Cascavilla, G., Tamburri, D. A., and Van Den Heuvel, W.-J. (2021). Cybercrime threat intelligence: A systematic multi-vocal literature review. *Computers & Security*, 105:102258.
- Celestini, A., Me, G., and Mignone, M. (2016). Tor marketplaces exploratory data analysis: The drugs case. pages 218–229.
- Choi, D., Ko, B., Kim, H., and Kim, P. (2014). Text analysis for detecting terrorism-related articles on the web. *Journal of Network and Computer Applications*, 38:16–21.
- Dalins, J., Wilson, C., and Carman, M. (2018). Criminal motivation on the dark web: A categorisation model for law enforcement. *Digital Investigation*.
- Décary-Héту, D., Mousseau, V., and Vidal, S. (2018). Six years later: Analyzing online black markets involved in herbal cannabis drug dealing in the united states. *Contemporary Drug Problems*, 45(4):366–381.
- Engebretson, P. (2013). Chapter 2 - reconnaissance. In Engebretson, P., editor, *The Basics of Hacking and Penetration Testing (Second Edition)*, pages 19 – 51.
- Graczyk, M. and Kinningham, K. (2015). Automatic product categorization for anonymous marketplaces.
- Hajba, G. L. (2018). Using beautiful soup. In *Website Scraping with Python*, pages 41–96. Springer.
- Mansfield-Devine, S. (2014). Tor under attack. *Computer Fraud & Security*, 2014(8):15 – 18.
- Minnaar, A. (2017). Online ‘underground’ marketplaces for illicit drugs: The prototype case of the dark web website ‘silk road’. page 2017.
- Sabbah, T., Selamat, A., Selamat, M. H., Ibrahim, R., and Fujita, H. (2015). Hybridized term-weighting method for dark web classification. *Neurocomputing*, 173.
- Spitters, M., Klaver, F., Koot, G., and Staalduinen, M. (2015). Authorship analysis on dark marketplace forums.
- Tavabi, N., Bartley, N., Abeliuk, A., Soni, S., Ferrara, E., and Lerman, K. (2019). Characterizing activity on the deep and dark web. In *WWW ’19*.
- Tsai, C.-F. (2012). Bag-of-words representation in image annotation: A review. *International Scholarly Research Notices*, 2012.
- Uysal, A. K. and Gunal, S. (2014). The impact of preprocessing on text classification. *Information Processing & Management*, 50(1):104–112.
- Vijayarani, S., Ilamathi, M. J., and Nithya, M. (2015). Pre-processing techniques for text mining-an overview. *International Journal of Computer Science & Communication Networks*, 5(1):7–16.
- Yang, L., Liu, F., Kizza, J., and Ege, R. (2009). Discovering topics from dark websites. pages 175 – 179.
- Yun-tao, Z., Ling, G., and Yong-cheng, W. (2005). An improved tf-idf approach for text classification. *Journal of Zhejiang University-Science A*, 6(1):49–55.