Intrusion Detection and Datasets

Subjects: Computer Science, Artificial Intelligence

Contributor: Joaquín Gaspar Medina-Arco , Roberto Magán-Carrión , Rafael Alejandro Rodríguez-Gómez , Pedro García-Teodoro

With the significant increase in cyber-attacks and attempts to gain unauthorised access to systems and information, Network Intrusion-Detection Systems (NIDSs) have become essential detection tools. Anomaly-based systems use machine learning techniques to distinguish between normal and anomalous traffic. They do this by using training datasets that have been previously gathered and labelled, allowing them to learn to detect anomalies in future data. However, such datasets can be accidentally or deliberately contaminated, compromising the performance of NIDS.

anomaly detection NIDS deep learning datasets network traffic labelling

1. Introduction

Network Intrusion-Detection Systems (NIDSs) represent a primary cybersecurity mechanism for identifying potential attacks on a communication network. To accomplish this goal, they analyse the network traffic passing through the system, regardless of whether it is internally generated or originated from external entities targeting the network. Detecting intrusions allows network administrators to become aware of system vulnerabilities and to make quick decisions to abort or mitigate attacks. Additionally, NIDSs allow them to implement measures to strengthen the system in the future [1].

NIDSs can be categorised into various typologies based on two fundamental principles: architecture and techniques employed. Focusing on the architecture, NIDS can be classified as host-based, network-based, and collaborative approaches between different components. According to the detection technique, the classification may be signature-based, Stateful Protocol Analysis-based, or anomaly detection-based NIDSs [2].

Signature-based NIDSs possess a repository of network patterns representing prevalent network attacks. Their operating mode is to match the network sequence they examine with their knowledge base to detect potential attacks [3].

Alternatively, Stateful Protocol Analysis-based NIDSs rely on their comprehensive understanding of the monitored protocol. They analyse all interactions to identify a sequence of actions that might result in a vulnerability or insecurity [3].

In contrast, anomaly-detection-based NIDSs employ mechanisms to detect abnormal network traffic behaviour. These anomalous activities typically correspond to network traffic patterns that have a significantly low likelihood of occurring or are markedly misaligned with normal traffic. Acutely objective, anomaly detection allows for the handling of novel or previously unknown attacks (*zero days*). This is because such attacks generate traffic patterns that have not been found before, and this type of NIDS often relies on the use of machine learning techniques to carry out anomaly detection. When this approach is followed, the subjective evaluation of attacks is effectively circumvented.

Different strategies have been employed to detect anomalies in NIDS through various machine learning techniques [4][5], including statistical techniques like Principal Component Analysis (PCA) [6] or Markov models [7][8]; classification techniques like Artificial Neural Networks (ANNs) [9][10][11][12], Support Vector Machines (SVMs) [6], deep learning models [13][14] including Autoencoders [9][15], or Decision Trees including Random Forest [16]; and clustering like outlier detection [17]. Using these techniques requires a multi-perspective approach to tackling the problem, which can be categorised as supervised, semi-supervised, or unsupervised, depending on the specific technique chosen [18].

Regardless of the technique used for anomaly detection in NIDS, the underlying models must be trained to distinguish normal traffic from anomalous traffic. This training process utilises datasets comprising real, synthetic, or a combination of both network traffic. To be more concise,

- **Synthetic traffic datasets** are created by generating traffic in a controlled environment that emulates a real-world setting. The generated traffic may include traffic related to known attacks, providing enough samples for machine learning models to competently identify and detect such anomalies. This enables the optimisation of the dataset regarding the size and balance between regular and irregular traffic samples. It also ensures the correct labelling of each observation as it has been intentionally and deliberately generated. Such observations can be, for instance, the traffic flows seen in the network. However, a potential issue is that it may not accurately reflect the network traffic patterns observed in a genuine environment.

- **Real traffic datasets** capture all network communications within a real productive environment. This implies access to the patterns of network traffic consumption and usage that take place in an actual scenario and potentially any cyber-attacks that may occur. Unlike synthetic datasets, real traffic samples may be biased or imbalanced, with the presence of anomalous traffic often being minimal or completely absent. It is necessary to carry out a subsequent process to assign a normality or attack label to each flow for its use in machine learning models during training phases.

- **Composite datasets** are the ones generated by combining real environment data and synthetic traffic to introduce attack patterns.

Regardless of the AI model used in a NIDS, the dataset's labelling accuracy is crucial to maintaining high model performance. This principle applies equally to supervised and unsupervised learning. In supervised learning,

labelling is necessary to enable models to learn how to identify anomalous traffic. In contrast, unsupervised learning generally assumes that the training dataset consists of normal traffic only and is, therefore, free of anomalies.

2. Datasets for Network Security Purposes

To effectively train any AI model, especially those constituting NIDSs based on anomaly detection, a prerequisite is a comprehensive dataset. This dataset should encompass a sufficient number of samples that represent all the various classes or patterns, whether benign or malicious. This foundational dataset enables the model to learn and predict accurately during subsequent training phases. In the specific case of NIDSs, a large and correctly labelled dataset is assumed [19]. The quality of the trained models depends to some extent on the quality of the data on which they were trained [20], so it is important to make a thorough analysis of the typology of datasets available in the NIDS domain.

Before reviewing the different datasets available in the field of cybersecurity, it is necessary to define the criteria according to which these datasets will be analysed:

- **Availability**: Understood as free access (Public) to the dataset or, on the contrary, of reserved access, by means of payment or explicit request (Protected).
- **Collected data**: Some datasets collect traffic packet for each packet (e.g., PCAP files), others collect information associated with traffic flows between devices (e.g., NetFlow), and others extract features from the flows by combining them with data extracted from the packets.
- **Labelling**: This refers to whether each observation in the dataset has been identified as normal, anomalous, or even belonging to a known attack. Or, conversely, no labelling is available, in which case they are intended for unsupervised learning models.
- **Type**: The nature of a dataset may be synthetic, where the process and environment in which the dataset is generated are controlled, or it may be the result of capturing traffic in a real environment.
- **Duration**: Network traffic datasets consist of network traffic recorded over a specific time interval, which may range from hours to days, months, or even years.
- **Size**: the depth of the dataset in terms of the number of records or the physical size and their distribution across the different classes.
- **Freshness**: It is also important to consider the year in which the dataset was created, as the evolution of attacks and network usage patterns may not be reflected in older datasets, thus compromising their validity in addressing current issues.

A summary of the datasets analysed according to the characteristics described above is shown in **Table 1**.

Table 1. Overview of available network datasets.

Dataset	Availability	Collected Data	Labeled	Type	Duration *	Size **	Year	Freshness	Balanced
DARPA [21]	Public	packets	yes	synthetic	7 weeks	6.5TB	1998–1999	questioned	no
NSL-KDD [22]	Public	features	yes	synthetic	N.S.	5M o.	1998–1999	questioned	yes
Kyoto 2006+ [23]	Public	features	yes	real	9 years	93M o.	2006–2015	yes	yes
Botnet [24]	Public	packets	yes	synthetic	N.S.	14GB p.	2010–2014	yes	yes
UNSW-NB15 [25]	Public	features	yes	synthetic	31 hours	2.5M o.	2015	yes	no
UGR'16 [26]	Public	flows	yes	real	6 months	17B f.	2016	yes	no
CICIDS2017 [27]	Protected	flows	yes	synthetic	5 days	3.1M f.	2017	yes	no
IDS2018 [28]	Protected	features	yes	synthetic	10 days	1M o.	2018	yes	no
NF-UQ-NIDS [29]	Public	flows	yes	synthetic	N.S.	12M f.	2021	yes	no

* N.S. means not specified. ** Expressed in flows (f.), observations (o.), or packets (p.). An observation denotes a data point with all specified features.

2.1. DARPA Datasets

Created by MIT's Lincoln Laboratory, the DARPA datasets, with KDD datasets, are perhaps the most widely used in the field of intrusion-detection systems [30]. There are two versions, one created in 1998 and the other in 1999. Both collect synthetically generated network packets in controlled network environments simulating network traffic patterns previously observed in production environments. In the case of the 1998 version, the duration of the training subset is seven weeks of data, while in the 1999 version, the training subset consists of only three weeks of observations. In both cases, two weeks of observed network traffic is reserved for validation. All observations are labelled and contain a total of 200 observations of up to 58 attacks of different typologies, including different versions of denial of service (DoS), port scanning, and user-to-root (U2R) or remote-to-local attacks (R2L) [21].

References

1. Ahmad, Z.; Shahid Khan, A.; Wai Shiang, C.; Abdullah, J.; Ahmad, F. Network intrusion detection system: A systematic study of machine learning and deep learning approaches. Trans. Emerg. Telecommun. Technol. 2021, 32, e4150.
2. Liao, H.J.; Richard Lin, C.H.; Lin, Y.C.; Tung, K.Y. Intrusion detection system: A comprehensive review. J. Netw. Comput. Appl. 2013, 36, 16–24.
3. Murali, A.; Rao, M. A Survey on Intrusion Detection Approaches. In Proceedings of the 2005 International Conference on Information and Communication Technologies, Karachi, Pakistan, 27–28 August 2005; pp. 233–240.
4. Patcha, A.; Park, J.M. An overview of anomaly detection techniques: Existing solutions and latest technological trends. Comput. Netw. 2007, 51, 3448–3470.

These datasets, despite the years they were built, are still used today in various scenarios and their usefulness seems to be proven [31], although there are some authors that question their utility [32].

2.2. KDD Dataset

KDD99 [22] is a dataset created for the Third International Knowledge Discovery and Data Mining Tools Competition based on the DARPA dataset. Unlike the latter, KDD99 is a dataset whose format is based on the extraction of features (up to 41 [33]) from network flows rather than the recording of raw observed data. It is a synthetic dataset but takes into account the actual traffic observed in military network environments. Access to the dataset is open, and, despite its longevity, it is still available. In terms of size, the dataset contains almost 5 million observations, including the same typology of attacks as DARPA, i.e., DoS, port scanning and privilege escalation attacks.

Similar to DARPA, although it is a widely employed dataset, criticisms have emerged regarding its usability. Specifically, concerns have been raised about the lack of consistency between the number of attack types in the training subset and those available in the validation subset [34]. Additionally, the dataset is deemed outdated in the context of contemporary world communications.

2.3. NSL-KDD Dataset

In 2009, to reduce the original DARPA and KDD problems, Tavallaee et al. [23] created a new version of KDD called NSL-KDD [23]. In this version, the authors removed all redundant records and added new synthetic ones based on the correctly labelled records of the original dataset, so that those record types with a lower presence in the original dataset had a higher presence in the new dataset and vice versa. As for the test dataset, it was completely regenerated. The result is a public dataset that is slightly more balanced, but with a very significant reduction in size, with just over 125 K observations in the training and 22.5 K in the testing set.

Even with the revision of the KDD dataset and the application of techniques to rebalance and address consistency issues, it continues to share the problems of its KDD and DARPA predecessors. Specifically, it relies on 1998 network traffic, rendering it outdated in the context of modern network communications and contemporary cyber-attacks.

2.4. Kyoto 2006+ Dataset

Given the shortcomings of datasets such as DARPA and KDD with their variants related to the longevity of their data, in 2006, Song et al. [35] published a new dataset called Kyoto 2006+, the result of recording real traffic from 32 honeypots with different characteristics from November 2006 to August 2009 (almost three years), totalling more than 93 million observations [35]. Since its initial publication, the authors have expanded the dataset to cover a total of nine years of traffic (up to 2015), adding more honeypots to reach the final figure of 348, including DNS servers to generate benign traffic. Each record in the dataset provides a total of 24 features associated with the captured network traffic flows, of which a total of 14 are present in datasets such as DARPA or KDD, while the remaining 10 are new additions, including the labelling of the records, as well as the typology of the detected

5. García-Teodoro, P.; Díaz-Verdejo, J.; Maciá-Fernández, G.; Vázquez, E. Anomaly-based network intrusion detection: Techniques, systems and challenges. Comput. Secur. 2009, 28, 18–28.
6. Wang, H.; Gu, J.; Wang, S. An effective intrusion detection framework based on SVM with feature augmentation. Knowl.-Based Syst. 2017, 136, 130–139.
7. Yeung, D.Y.; Ding, Y. Host-based intrusion detection using dynamic and static behavioral models. Pattern Recognit. 2003, 36, 229–243.
8. Mahoney, M.V.; Chan, P.K. Learning nonstationary models of normal network traffic for detecting novel attacks. In Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Edmonton, AB, Canada, 23–26 July 2002; pp. 376–385.
9. Mirsky, Y.; Doitshman, T.; Elovici, Y.; Shabtai, A. Kitsune: An Ensemble of Autoencoders for Online Network Intrusion Detection. arXiv 2018, arXiv:1802.09089.
10. Li, J.; Manikopoulos, C.; Jorgenson, J.; Ucles, J. HIDE: A Hierarchical Network Intrusion Detection System Using Statistical Preprocessing and Neural Network Classification. In Proceedings of the 2001 IEEE Workshop on Information Assurance and Security, West Point, NY, USA, 5–6 June 2001.
11. Poojitha, G.; Kumar, K.N.; Reddy, P.J. Intrusion Detection using Artificial Neural Network. In Proceedings of the 2010 Second International Conference on Computing, Communication and Networking Technologies, Karur, India, 29–31 July 2010; pp. 1–7.
12. Ghosh, A.K.; Michael, C.; Schatz, M. A Real-Time Intrusion Detection System Based on Learning Program Behavior. In Recent Advances in Intrusion Detection, Proceedings of the Third International Workshop, RAID 2000, Toulouse, France, 2–4 October 2000; Debar, H., Mé, L., Wu, S.F., Eds.; Lecture Notes in Computer Science; Springer: Berlin/Heidelberg, Germany, 2000; pp. 93–109.
13. Ullah, S.; Ahmad, J.; Khan, M.A.; Alkhammash, E.H.; Hadjouni, M.; Ghadi, Y.Y.; Saeed, F.; Pitropakis, N. A New Intrusion Detection System for the Internet of Things via Deep Convolutional Neural Network and Feature Engineering. Sensors 2022, 22, 3607.
14. Banaamah, A.M.; Ahmad, I. Intrusion Detection in IoT Using Deep Learning. Sensors 2022, 22, 8417.
15. Ren, Y.; Feng, K.; Hu, F.; Chen, L.; Chen, Y. A Lightweight Unsupervised Intrusion Detection Model Based on Variational Auto-Encoder. Sensors 2023, 23, 8407.
16. Kotecha, K.; Verma, R.; Rao, P.V.; Prasad, P.; Mishra, V.K.; Badal, T.; Jain, D.; Garg, D.; Sharma, S. Enhanced Network Intrusion Detection System. Sensors 2021, 21, 7835.
17. Chandola, V.; Eilertson, E.; Ertoz, L.; Simon, G.; Kumar, V. Minds: Architecture & Design. In Data Warehousing and Data Mining Techniques for Cyber Security; Singhal, A., Ed.; Advances in

attack. This dataset is probably the public dataset of real traffic with the greatest historical depth on record, but, in spite of this, it is still quite balanced.

2.5. Botnet Dataset

Biglar Beigi et al. [24] have developed a public dataset focused on botnet attacks, as they believed that this type of attack is currently the most challenging [24]. This dataset contains a total of 16 different botnet attack typologies, covering both centralised and decentralised attack strategies. In order to construct this synthetic dataset, the researchers analysed different datasets by combining subsets of three different datasets (ISOT [36], ISCX 2012 IDS dataset [37], and *Botnet Traffic Generated by the Malware Capture Facility Project* or CTU-13 [38]) using the overlay methodology described in [39] that ensures the cohesion of the resulting data. The result is a dataset of tagged network packets with a total of almost 14 GB of information and a balance between normal and anomalous traffic of almost 55% and 45%, respectively, which is quite balanced.

2.6. UNSW-NB15

The Cyber Range Lab at the Australian Centre for Cyber Security generated the synthetic UNSW-NB15 [25] dataset in 2015 using the IXIA Perfect Storm traffic generator. The simulation environment used to generate the samples consists of three servers, two of which generate benign traffic, while the third is used to generate traffic associated with various attacks such as DoS, exploits, and rootkits. The dataset size is reduced, reaching a total of 31 h in two subsets of 16 and 15 h, respectively, with just under 2.5 million observations, 12% of which correspond to anomalies or attacks. Labels are available for each flow, indicating whether it is normal or not, as well as the attack category to which it belongs. Finally, the data are available in packet format (PCAP) as a version of 49 features extracted from the captured flows.

2.7. UGR’16

The UGR’16 dataset [26] was created by the University of Granada in 2016 as a result of capturing the real network traffic of a medium-sized ISP between March and June 2016. Subsequently, during the months of July and August, different attacks such as DoS, botnet, or port scanning were deliberately generated on the split ISP to capture all the traffic so that this subset could be used as a test. The dataset consists of NetFlow traffic flows with almost 17 billion different connections, of which more than 98% were normal traffic, making it very imbalanced. After the traffic was captured, state-of-the-art anomaly detection and network attack identification techniques were employed to tag the dataset. This involved assigning each record a label indicating the type of attack to which it belonged. Given the size of the dataset and its temporal proximity, it is an updated and current dataset for use in building or training AI and NIDS models.

2.8. CIC Datasets

The Canadian Institute for Cybersecurity (CIC) has generated several datasets to validate the performance of NIDS or to train the models underlying these NIDS. Among the various datasets available, the following should be

Information Security; Springer: Boston, MA, USA, 2007; pp. 83–107.
18. Ahmed, M.; Naser Mahmood, A.; Hu, J. A survey of network anomaly detection techniques. J. Netw. Comput. Appl. 2016, 60, 19–31.
19. De Keersmaeker, F.; Cao, Y.; Ndonda, G.K.; Sadre, R. A Survey of Public IoT Datasets for Network Security Research. IEEE Commun. Surv. Tutor. 2023, 25, 1808–1840.
20. Camacho, J.; Wasielewska, K.; Espinosa, P.; Fuentes-García, M. Quality In/Quality Out: Data quality more relevant than model choice in anomaly detection with the UGR’16. In Proceedings of the NOMS 2023—2023 IEEE/IFIP Network Operations and Management Symposium, Miami, FL, USA, 8–12 May 2023; pp. 1–5.
21. Lippmann, R.; Haines, J.W.; Fried, D.J.; Korba, J.; Das, K. The 1999 DARPA off-line intrusion detection evaluation. Comput. Netw. 2000, 34, 579–595.
22. Salvatore Stolfo, W.F. KDD Cup 1999 Data; UCI Machine Learning Repository, 1999.
23. Tavallaee, M.; Bagheri, E.; Lu, W.; Ghorbani, A.A. A detailed analysis of the KDD CUP 99 data set. In Proceedings of the 2009 IEEE Symposium on Computational Intelligence for Security and Defense Applications, Ottawa, ON, Canada, 8–10 July 2009; pp. 1–6.
24. Biglar Beigi, E.; Hadian Jazi, H.; Stakhanova, N.; Ghorbani, A.A. Towards effective feature selection in machine learning-based botnet detection approaches. In Proceedings of the 2014 IEEE Conference on Communications and Network Security, San Francisco, CA, USA, 29–31 October 2014; pp. 247–255.
25. Moustafa, N.; Slay, J. UNSW-NB15: A comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set). In Proceedings of the 2015 Military Communications and Information Systems Conference (MilCIS), Canberra, ACT, Australia, 10–12 November 2015; pp. 1–6.
26. Maciá-Fernández, G.; Camacho, J.; Magán-Carrión, R.; García-Teodoro, P.; Therón, R. UGR‘16: A new dataset for the evaluation of cyclostationarity-based network IDSs. Comput. Secur. 2018, 73, 411–424.
27. Sharafaldin, I.; Habibi Lashkari, A.; Ghorbani, A.A. Toward Generating a New Intrusion Detection Dataset and Intrusion Traffic Characterization. In Proceedings of the 4th International Conference on Information Systems Security and Privacy, Funchal, Madeira, Portugal, 22–24 January 2018; pp. 108–116.
28. Canadian Institute for Cybersecurity. CSE-CIC-IDS2018. 2018. Available online: https://www.unb.ca/cic/datasets/ids-2018.html (accessed on 30 November 2023).
29. Sarhan, M.; Layeghy, S.; Moustafa, N.; Portmann, M. NetFlow Datasets for Machine Learning-Based Network Intrusion Detection Systems. In Big Data Technologies and Applications,

highlighted.

- CICIDS2017 [27]: Generated in 2017, it is a synthetic network traffic dataset generated in a controlled environment for a total of 5 days, available on request (it is protected). The captured data are in packet and flow formats, although they are also available in extracted feature format with a total of 80 different features. The captured traffic is tagged, and the different attacks that each record corresponds to, including DoS, SSH, and botnet attacks, are marked in the tag.
- CSE-CIC-IDS2018 [28]: This is a synthetic dataset generated in 2018 specifically based on network traffic intrusion criteria. It includes DoS attacks, web attacks and network infiltration, among others, recorded on more than 400 different hosts. As with CICIDS2017, the data are in packet and flow format, but with a version containing 80 extracted features, and access requires a prior request (protected). Unlike CICIDS2017, it is modifiable and extensible.

2.9. NF-UQ-NIDS

Sarhan et al. [29] have created a synthetic dataset specifically created for machine learning-based NIDSs [29]. This dataset is the result of combining four datasets used in the NIDS domain but transformed into a netflow version. Two of the datasets used have been analysed previously in this research (UNSW-NB15 [25] and CSE-CIC-IDS2018 [28]), while the other two (BoT-IoT [40] and ToN-IoT [41]) are datasets generated by the Cyber Range Lab of the Australian Centre for Cyber Security (ACCS). The result is a dataset that contains flows from different networks with different configurations, making it more universal than the datasets of which it is composed. The original dataset to which each flow belongs is available, allowing us to know under which scenario or network a NIDS trained with NF-UQ-NIDS can be more or less effective. The dataset contains almost 12 M records, 76.77% of which correspond to normal traffic, while the remaining 23.33% correspond to the 20 types of attacks it contains, making it an imbalanced dataset. It was published in 2021, so it is a dataset that can be considered up-to-date and incorporates the latest types of attacks.

3. Dealing with Labelling Problems in Datasets and the Techniques to Address Them

Classification problems, whether supervised or unsupervised learning, depend on a sufficiently large dataset that is correctly labelled. In the case of supervised learning, the labelling is used so that the model learns to distinguish the different classes that make up the universe being treated. However, when the problem is approached from an unsupervised learning perspective, such as anomaly detection, the training dataset is expected to belong to the same class. This setup enables the model to learn to identify anomalies by recognizing deviations from the patterns present in the training set. The process of creating a dataset is therefore very important, as it determines the potential success of the machine learning models that will use it.

Proceedings of the 10th EAI International Conference, BDTA 2020, and 13th EAI International Conference on Wireless Internet, WiCON 2020, Virtual Event, 11 December 2020; Deze, Z., Huang, H.; Hou, R.; Rho, S.; Chilamkurti, N., Eds.; Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering; Springer: Cham, Switzerland, 2021; pp. 117–135.

30. Ring, M.; Wunderlich, S.; Scheuring, D.; Landes, D.; Hotho, A. A Survey of Network-based Intrusion Detection Data Sets. Comput. Secur. 2019, 86, 147–167.
31. Thomas, C.; Sharma, V.; Balakrishnan, N. Usefulness of DARPA dataset for intrusion detection system evaluation. In Proceedings of the Data Mining, Intrusion Detection, Information Assurance, and Data Networks Security, Orlando, FL, USA, 17–18 March 2008; Volume 6973, pp. 164–171.
32. McHugh, J. Testing Intrusion detection systems: A critique of the 1998 and 1999 DARPA intrusion detection system evaluations as performed by Lincoln Laboratory. ACM Trans. Inf. Syst. Secur. 2000, 3, 262–294.
33. Chaabouni, N.; Mosbah, M.; Zemmari, A.; Sauvignac, C.; Faruki, P. Network Intrusion Detection for IoT Security Based on Learning Techniques. IEEE Commun. Surv. Tutor. 2019, 21, 2671–2701.
34. Sabahi, F.; Movaghar, A. Intrusion Detection: A Survey. In Proceedings of the 2008 Third International Conference on Systems and Networks Communications, Sliema, Malta, 26–31 October 2008; pp. 23–26.
35. Song, J.; Takakura, H.; Okabe, Y.; Eto, M.; Inoue, D.; Nakao, K. Statistical analysis of honeypot data and building of Kyoto 2006+ dataset for NIDS evaluation. In Proceedings of the First Workshop on Building Analysis Datasets and Gathering Experience Returns for Security, BADGERS '11, Salzburg, Austria, 10 April 2011; pp. 29–36.
36. Saad, S.; Traore, I.; Ghorbani, A.; Sayed, B.; Zhao, D.; Lu, W.; Felix, J.; Hakimian, P. Detecting P2P botnets through network behavior analysis and machine learning. In Proceedings of the 2011 Ninth Annual International Conference on Privacy, Security and Trust, Montreal, QC, Canada, 19–21 July 2011; pp. 174–180.
37. Shiravi, A.; Shiravi, H.; Tavallaee, M.; Ghorbani, A.A. Toward developing a systematic approach to generate benchmark datasets for intrusion detection. Comput. Secur. 2012, 31, 357–374.
38. García, S.; Grill, M.; Stiborek, J.; Zunino, A. An empirical comparison of botnet detection methods. Comput. Secur. 2014, 45, 100–123.
39. Aviv, A.J.; Haeberlen, A. Challenges in experimenting with botnet detection systems. In Proceedings of the 4th Conference on Cyber Security Experimentation and Test, San Francisco, CA, USA, 8 August 2011; CSET'11. p. 6.

The process of tagging the data that make up a dataset involve the application of automated techniques, as well as manual processes, which together can be subject to error [42]. To mitigate this problem, some papers present methods or techniques to reduce the mislabelling that occurs. For example, Kremer et al. [43] propose a model that tries to detect the noise in the labelling based on loss functions that are insensitive to noise and at the same time tries to infer the possible noise in the labelling and in the classification itself [43]. On the other hand, Zhang et al. [44] propose a framework called Adaptive Voting Noise Correction (AVNC), which aims to identify and correct incorrect labelling [44]. However, even the application of these techniques does not guarantee the correct labelling of the dataset.

When the labelling of the data that make up a dataset is performed manually, there is a risk of unintentional bias that is intrinsic to the observed data. To address this scenario, a methodology is proposed in [45], whose aim is to relabel the data, eliminating the possible bias of the initial labelling, achieving good results in a computational perception problem on galaxy detection.

The impact of noise on labelling in artificial intelligence models has also been analysed in several works in a way that relativises its impact. For example, Natarajan et al. [46] propose in [46] a simple loss estimator that is unbiased and minimises the risk of the presence of mislabelled data. Another approach, as proposed by Patrini et al. [47], focuses on tackling the issue of noise in labelling, particularly in scenarios involving deep learning models, including recurrent neural networks. The researchers suggest two procedures to correct the loss function in instances of mislabelled data [47]. More recent is the work of Wei et al. [48], who this problem and propose two datasets with noise in the labelling to serve as a benchmark to measure how robust the models or techniques are to errors in the labelling [48].

Of particular relevance is the work of Northcutt et al. [42], which analyses the quality of labelling in test subsets of 10 datasets, as opposed to the work presented above, which focuses on the quality of labelling of the training data. This approach is particularly interesting as the test subsets are assumed to be perfectly labelled, as they are the test and evaluation mechanism by which the models are tested and validated [42]. Labelling errors in such a dataset can destabilise the performance of machine learning models. The datasets tested are those commonly used in the field of computational perception (such as MNIST or ImageNet), in the field of language processing (such as IMDB or Amazon Reviews), and finally in the field of audio processing (AudioSet). The results obtained show that there are labelling errors that, in some cases, reach up to 10% of the labelling error.

Confident Learning (CL) is a subfield of machine learning between supervised and semi-supervised learning that focuses on characterising noise in the labelling to find and correct errors in the labelling in order to train robust models. To achieve this, they use data-pruning techniques to clean the dataset before training the models. In [49], a generalised CL strategy is proposed that is able to find the errors in the labelling by estimating the correct distribution of correct and incorrect labels. Furthermore, it is tested on image datasets, yielding models with higher performance than some of the best state-of-the-art models.

40. Koroniotis, N.; Moustafa, N.; Sitnikova, E.; Turnbull, B. Towards the Development of Realistic Botnet Dataset in the Internet of Things for Network Forensic Analytics: Bot-IoT Dataset. arXiv 2018, arXiv:1811.00701.
41. Moustafa, N. ToN_IoT Datasets; IEEE: Piscataway, NJ, USA, 2019.
42. Northcutt, C.G.; Athalye, A.; Mueller, J. Pervasive Label Errors in Test Sets Destabilize Machine Learning Benchmarks. arXiv 2021, arXiv:2103.14749.
43. Kremer, J.; Sha, F.; Igel, C. Robust Active Label Correction. In Proceedings of the Twenty-First International Conference on Artificial Intelligence and Statistics. PMLR, Playa Blanca, Spain, 9–11 April 2018; pp. 308–316.
44. Zhang, J.; Sheng, V.S.; Li, T.; Wu, X. Improving Crowdsourced Label Quality Using Noise Correction. IEEE Trans. Neural Netw. Learn. Syst. 2018, 29, 1675–1688.
45. Cabrera, G.F.; Miller, C.J.; Schneider, J. Systematic Labeling Bias: De-biasing Where Everyone is Wrong. In Proceedings of the 2014 22nd International Conference on Pattern Recognition, Stockholm, Sweden, 24–28 August 2014; pp. 4417–4422.
46. Natarajan, N.; Dhillon, I.S.; Ravikumar, P.K.; Tewari, A. Learning with Noisy Labels. In Advances in Neural Information Processing Systems; Curran Associates, Inc.: Red Hook, NY, USA, 2013; Volume 26.
47. Patrini, G.; Rozza, A.; Menon, A.; Nock, R.; Qu, L. Making Deep Neural Networks Robust to Label Noise: A Loss Correction Approach. arXiv 2017, arXiv:1609.03683.
48. Wei, J.; Zhu, Z.; Cheng, H.; Liu, T.; Niu, G.; Liu, Y. Learning with Noisy Labels Revisited: A Study Using Real-World Human Annotations. arXiv 2022, arXiv:2110.12088.
49. Northcutt, C.; Jiang, L.; Chuang, I. Confident Learning: Estimating Uncertainty in Dataset Labels. J. Artif. Intell. Res. 2021, 70, 1373–1411.
50. Müller, N.M.; Markert, K. Identifying Mislabeled Instances in Classification Datasets. In Proceedings of the 2019 International Joint Conference on Neural Networks (IJCNN), Budapest, Hungary, 14–19 July 2019; pp. 1–8.
51. Hao, D.; Zhang, L.; Sumkin, J.; Mohamed, A.; Wu, S. Inaccurate Labels in Weakly-Supervised Deep Learning: Automatic Identification and Correction and Their Impact on Classification Performance. IEEE J. Biomed. Health Inform. 2020, 24, 2701–2710.
52. Bekker, A.J.; Goldberger, J. Training deep neural-networks based on unreliable labels. In Proceedings of the 2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Shanghai, China, 20–25 March 2016; pp. 2682–2686.
53. Cordero, C.G.; Vasilomanolakis, E.; Wainakh, A.; Mühlhäuser, M.; Nadjm-Tehrani, S. On Generating Network Traffic Datasets with Synthetic Attacks for Intrusion Detection. ACM Trans.

Priv. Secur. 2021, 24, 1–39.
54. Guerra, J.L.; Catania, C.; Veas, E. Datasets are not enough: Challenges in labeling network traffic. Comput. Secur. 2022, 120, 102810.
55. Soukup, D.; Tisovčík, P.; Hynek, K.; Čejka, T. Towards Evaluating Quality of Datasets for Network Traffic Anomaly Detection: Sufficient Size and Commonness. In Proceedings of the 2021 17th International Conference on Network and Service Management (CNSM), Izmir, Turkey, 25–29 October 2021; pp. 264–268.

Müller and Markert [50] propose a tool to detect errors in the labelling of image, text and numerical datasets [50]. As a result of the application of this tool, the set of observations of the dataset with a high probability of being mislabelled is obtained. This method has been tested on a total of 29 different datasets, both real and synthetic and, according to its authors, has been able to find mislabelling in some of them that had not been detected before.

The application of computational techniques in medicine is also subject to the risk of dataset with mislabelling, especially when the goal is to detect outbreaks [51], proposed solutions. In [51], the researchers addressed this problem by proposing a methodology to identify labelling errors in images associated with the presence of breast cancer. To achieve this, they propose a function that measures the deviation between the prediction made by the model and the real value of the sample (called Cross-Entropy loss). Additionally, they put forward another function that assesses the model's dependence on the dataset, known as the Influence function. The method is evaluated on a set of 10,500 images in which up to 98% of labelling errors are detected.

Another methodology in the field of image processing is proposed in [52], where the aim is to train a deep learning model with a dataset where there is no confidence in the labelling of the data. To do this, the model adjusts the internal parameters of the neural network while learning the distribution of noise in the labelling and testing it against classical back-propagation models where the goodness of the labelling is assumed.

In the specific area of datasets aimed at addressing cybersecurity or network traffic problems, previous work is more limited, as the generation of these datasets has additional complications with respect to the more general use cases. In [53], Cordero et al. [53] the problem is reviewed through a comprehensive analysis of various datasets intended for NIDS. The researchers put forth an enhancement to the Intrusion-Detection Dataset Toolkit (ID2T) dataset generation methodology. Subsequently, they evaluate the effectiveness of the proposed ID2T improvement by assessing datasets generated after its application.

The problem of labelling in the field of network traffic is more complex, since it requires specific low-level knowledge of the traffic in order to be able to correctly classify each flow. In [54], an analysis of the methods used for labelling this type of dataset, both automatic and manual, is carried out, identifying the weaknesses of each of the techniques along with their advantages and disadvantages.

Finally, to conclude this analysis of the state of the art in dataset quality, in [55], an approach to measuring the quality of a network traffic dataset is presented. This quality is used to compare two datasets, to decide if they are equivalent, or if a better quality dataset is found, whether or not it is appropriate to retrain the machine learning models. The proposal for measuring the quality of a dataset is based on the criteria: (i) completeness as the probability that a dataset record can occur in the domain of the machine learning model to be built and (ii) reliability as the probability of occurrence of misclassified or mislabelled data for each possible class. Based on these two criteria, the applicability of a network traffic dataset to a particular problem can be determined.