Ralf Benzmüller

Machine Learning und Virenschutz

Nützlich – aber noch viel zu lernen

Die Erkennung und Abwehr von Schadprogrammen hat über die Jahrzehnte immer mehr Bedeutung gewonnen. Im Laufe der Zeit wurden zahlreiche Schutztechnologien entwickelt. Auch die Künstliche Intelligenz und Maschinelles Lernen werden erfolgreich eingesetzt. Welche Chancen bieten sie und wo sind ihre Grenzen?

Künstliche Intelligenz (KI) und Maschinelles Lernen (ML) und ihre Möglichkeiten und Anwendungsfelder werden aktuell unter vielen Aspekten diskutiert. In den letzten Jahrzehnten machten die intelligenten Maschinen viele Fortschritte und erzielten verblüffende Ergebnisse. Deep Blue bekam sehr viel öffentliche Aufmerksamkeit als es am 10. Februar 1996 den damals amtierenden Schachweltmeister Garry Kasparov schlug. Auch IBMs Watson demonstrierte die Leistungsfähigkeit von Computersystemen, als es im Februar 2011 zwei Sieger des Ratespiels Jeopardy schlug. AlphaGo knüpfte im März 2016 daran an und bezwang Lee Sedol, einen der besten, professionellen Go-Spieler weltweit. Der zugrundeliegende Algorithmus AlphaGo wurde vom DeepMind-Team um AlphaGo-Zero ergänzt[1] und noch einmal generalisiert zu Alpha Zero[2]. In nur 4 Stunden entwickelte sich AlphaZero durch intensives Selbsttraining per Reinforcement Learning vom Anfänger zum Profi und schlug Stockfish, eins der besten Schachprogramme der Welt[3].

Angesichts dieser Leistungen liegt der Gedanke nahe, dass man mit den Verfahren der Künstlichen Intelligenz und des Maschinellen Lernens auch in anderen Bereichen ähnliche Durchbrüche erzielen kann. Auch in der IT-Sicherheit und der Erkennung und Abwehr von Schadprogrammen sind in den letzten Jahren viele Verfahren entwickelt und in Produkte integriert worden. Mit vollmundigen Werbeversprechen[4] wecken die aufstrebenden Anbieter Hoffnungen und sorgen für Diskussionen[5] um die Leistungsfähigkeit der neuen Verfahren[6]. Aber was können die automatischen Verfahren leisten? Die überzogenen Erwartungen wurden schon bald gedämpft[7]. Wir werfen einen Blick hinter die Kulissen der Künstlichen Intelligenz und des Maschinellen Lernens und werden sehen, dass sich die Magie von ML als solide Mathematik, Statistik mit großen Datenmengen und sehr viel Rechenleistung entpuppt.

1 Wie klug sind Maschinen – Einleitung und Begrifflichkeiten

Wenn man anfängt zu untersuchen, wie Künstliche Intelligenz und Machine Learning funktionieren, wird man mit vielen Begriffen konfrontiert, die in vielfältiger Weise in Verbindung stehen. Das beginnt mit dem Begriff **Künstliche Intelligenz**. Sie gilt als Forschungsbereich in der Informatik. Mangels einer einheitlichen Definition von Intelligenz lässt sich das Forschungsfeld nur so einschränken, dass es nicht um die Abarbeitung vorher festgelegter Befehlssequenzen geht. Ob und wie weitere Kriterien wie Verarbeitung von Sprache, Visuelle Erkennung, Kreativität, Schlussfolgern, allgemeine Lernfähigkeit etc. berücksichtigt werden, hängt von der jeweiligen Sichtweise ab. Viele dieser Aufgaben werden durch die Auswertung von Daten gelöst. Der Bereich Strong AI grenzt sich davon ab. Die datenbasierten Ansätze werden unter dem Begriff **Maschinelles Lernen** zusammengefasst. Lernen impliziert, dass Informationen und Rückmeldungen von außen dazu führen, dass eine Aufgabe besser gelöst wird. Als Kernaufgaben des Maschinellen Lernens gelten a) Vorhersagen treffen (z.B. über Eigenschaften oder Ereignisse), b) Ursachen finden und verstehen und c) Muster in oft ungeordneten und großen Datenbeständen erkennen[8]. Die Entwicklung von statistischen Methoden und mathematischen Verfahren zur Lösung die-

1 Silver et al. 2017: Mastering the game of Go without human knowledge. In Nature 550. S. 354ff. online: https://www.nature.com/articles/nature24270.epdf

2 Silver et al. 2017: Mastering Chess and Shogi by Self-Play with a General Reinforcement Learning Algorithm. Online: https://arxiv.org/pdf/1712.01815.pdf

3 Dockrill, Peter, 2017: In Just 4 Hours, Google's AI Mastered All The Chess Knowledge in History. In: Science Alert. Online: https://www.sciencealert.com/it-took-4-hours-google-s-ai-world-s-best-chess-player-deepmind-alphazero

4 Cylance Werbevideo auf Youtube: https://www.youtube.com/watch?v=lyWTVN4XKa0

5 Gottlieb, Carl Mai 2016: The AV Bomb That Never Was. Online: https://carl-gottlieb.com/av-virustotal-cylance/

6 Kubovic, April 2017: [1]

7 AV-Test Juli 2017: Full Product Test of Palo Alto Network Traps. Online: https://www.av-test.org/fileadmin/pdf/reports/AV-TEST_Palo_Alto_Traps_Full_Product_Test_July_2017.pdf

8 https://i.imgur.com/HnRwlce.png

Ralf Benzmüller

hat die G DATA SecurityLabs aufgebaut und bei der Entwicklung von Systemen zur automatischen Malware-Analyse auch intensiv mit Machine-Learning-Verfahren gearbeitet.
E-Mail: ralf.benzmueller@gdata.de

ser Aufgaben wird auch dem Maschinellen Lernen zugeordnet[9]. Die Anwendung dieser Methoden gilt als Gegenstand des neuen Felds der **Data Science**. Sie befasst sich mit den praktischen Herausforderungen bei der Erhebung, Aufbereitung, Auswertung und Darstellung von Daten[10]. Maschinelles Lernen wird bei der Auswertung der Daten eingesetzt.

Bevor wir uns der Anwendung des Maschinellen Lernens auf die Erkennung von Schadprogrammen zuwenden sollen noch einige grundlegende Methoden und Verfahren des Maschinellen Lernens vorgestellt werden.

2 Wie funktioniert Maschinelles Lernen?

Die Basis für die Lernfähigkeit von Maschinen sind Daten, die in ihrer Menge und ihrem Informationsgehalt geeignet sind, für eine Fragestellung Lösungen zu liefern. Mit der Qualität und der Menge der Daten steht und fällt die Qualität der Ergebnisse. Die Art und Weise wie die Daten von den Maschinen genutzt werden können, hängt davon ab, ob sie vorher (manuell) für die Lösung der Aufgabe bewertet und auf Korrektheit geprüft wurden.

2.1 Überwachtes Lernen

Liegen derart aufbereitete Daten vor, können Verfahren des Supervised Learning genutzt werden. Anhand von verfügbaren Merkmalen (Features) können per Regression Vorhersagen gebildet werden. Viel wichtiger für den Bereich der Malware-Erkennung ist die Einordnung in vorgegebene Kategorien. Im einfachsten Fall ist es die Einordnung in die Gruppen „schädlich" und „nicht schädlich". Eine komplexere Aufgabe wäre die Zuordnung zu verschiedenen Malware-Typen (z.B. Keylogger, Wurm, Virus, Ransomware, Backdoor). Dafür werden Verfahren wie Neuronale Netze unterschiedlicher Komplexität[11] (Deep Learning), Support Vector Machines (SVM), Entscheidungsbäume oder Naïve Bayes genutzt. Die vorliegenden Daten werden in ein Testset und ein Trainingsset aufgeteilt. Die Lernverfahren starten mit einem initialen Modell und gewichten die Eingabewerte in vielen Durchläufen so, dass die vorgegebenen Resultate möglichst genau erreicht werden. Je nach Datenmenge und Komplexität des Verfahrens kann das sehr viel Rechenleistung erfordern. Die anhand des Trainingssets gelernten Bewertungsmodelle werden dann mit dem Testset überprüft. Mit den so ermittelten Erkennungsraten und Fehlermatrizen lassen sich Aussagen über die Güte des berechneten Modells treffen. Diese gelten allerdings nur für die vorliegenden Daten. Wenn das Modell im Anwendungsfall auf unbekannte Daten trifft, können die Resultate abweichen.

2.2 Datenselbstbestimmung

Nicht immer liegen die aufbereiteten Daten die für die Nutzung von Supervised Learning notwendig sind in ausreichender Menge und Qualität vor. Die Erstellung der Datenkorpora z.B. für Handschriftenerkennung, Spracherkennung und autonomes Fahren ist zeitaufwändig und teuer. Für viele Anwendungsfelder liegen aber unstrukturierte Daten vor. Mit Verfahren des **Unsupervised Learning**[12] können Maschinen versuchen Muster und Struktur in den Daten zu finden und sie nach Ähnlichkeiten zusammenzufassen. Bei dieser Aufgabe des Clustering geht es vorrangig darum, robuste Abstandsmetriken zu entwickeln, um ähnliche Fälle in gleiche Gruppen einzusortieren. Dazu werden u.a. Verfahren wie K-means oder Hidden Markov Modelle (HMM) verwendet. Die verfügbaren Daten werden anhand ihrer Eigenschaften in iterativen Schritten in Gruppen zusammengefasst (sog. Cluster). Ob und wie diese Gruppen auf Kategorien bezogen sind, die für menschliche Wahrnehmung relevant sind, lässt kaum bestimmen. Unsupervised Learning eignet sich gut, um unüberschaubare Datenmengen auf die wichtigsten Eigenschaften und Kriterien zu reduzieren[13].

2.3 Verhalten verstärken

Das Mauerblümchen der ML-Ansätze war lange Zeit das **Reinforcement Learning**. Es basiert auf der Idee, dass auch Verhalten von Maschinen belohnt und damit verstärkt werden kann. Software-*Agenten* führen in einer *Umgebung* bestimmte *Aktionen* durch und erhalten eine *Belohnung*. Ziel des Agenten ist es, eine möglichst hohe Belohnung zu erhalten.

2.4 Ungereimtheiten aufspüren

Für die Erkennung der Aktivitäten von Schadprogrammen ist die **Anomalieerkennung** ein wichtiger Bereich des Maschinellen Lernens. Zunächst wird dabei ein Normalzustand mit üblichen Vorgängen erfasst. Im späteren Einsatz werden dann ungewöhnliche Aktionen identifiziert und als Warnmeldungen ausgegeben. So lassen sich Ausreißer im Datenbestand gut erfassen [14]. Bei den Verfahren werden je nach Datenlage überwachte und unüberwachte Verfahren eingesetzt. Für den Fall, dass ein Teil der Daten aufbereitet und verifiziert wurde, gibt es auch die Mischform das halb-überwachten Lernens. Anomalieerkennung ist wesentlicher Bestandteil der Auswertung von Logdateien von Netzwerk-Traffic, Systemaktivitäten und –zuständen wie sie in Unified Threat Management Systemen eingesetzt werden.

2.5 Durchbruch – was ist neu?

Viele Verfahren, die beim Maschinellen Lernen eingesetzt werden, sind im letzten Jahrtausend entstanden und wurden bereits in den 90er Jahren in der Sprachverarbeitung, der Spracherkennung und zur Handschriftenerkennung eingesetzt. Es gab aber einige Entwicklungen, die dazu beigetragen haben, dass Maschinelles Lernen in viele Bereiche der Wirtschaft, Forschung und des täglichen Lebens eingezogen ist. Die Grundlage dafür sind einerseits die verbesserte Rechenleistung u.a. durch schnellere Prozes-

9 Mayo Mai 2016: [2]

10 Mayo, März 2016: The Data Science Process, rediscoverd. online: https://www.kdnuggets.com/2016/03/data-science-process-rediscovered.html

11 Im Rahmen dieses Artikels kann nicht näher auf die einzelnen Varianten von Neuronalen Netzen eingegangen werden. Einen guten Überblick und Einstieg bietet Fjodor von Veen vom Asimov Institug: online: http://www.asimovinstitute.org/neural-network-zoo/

12 Gahramani, 2004: Unsupervised Learning. online: http://mlg.eng.cam.ac.uk/zoubin/papers/ul.pdf

13 Eine knappe Beschreibung der wichtigsten ML-Algorithmen findet sich in Ray, Sep 2017. Essentials of Machine Learning Algorithms (with Python and R Codes) https://www.analyticsvidhya.com/blog/2017/09/common-machine-learning-algorithms/

14 Hodge & Austin, 2004: A Survey of Outlier Detection Methodologies. online: http://eprints.whiterose.ac.uk/767/1/hodgevj4.pdf

soren und Fortschritte in der parallelen Verarbeitung von Daten mit GPUs.

Auch die Verfügbarkeit von Daten hat sich in den letzten Jahrzehnten deutlich verbessert[15]. Der Umgang mit Daten ist wesentlich freizügiger geworden. In den 90er Jahren wurden Korpora mit Datenmaterial für bestimmte Forschungsbereiche (z.B. Spracherkennung, Übersetzung) in streng kontrollierten Projekten mit viel Aufwand erstellt. Die Übertragung von Gesundheitsdaten, Sprachaufnahmen und Geo-Lokalisierungsdaten zur maschinellen Verarbeitung an zentrale Server ist mittlerweile Usus, ungeachtet der Diskussionen um Privatsphäre und Datenschutz, die mit der EU-Datenschutzgrundverordnung adressiert werden.

Wichtige Fortschritte wurden auch bei der Verarbeitung der Daten erzielt. Die Beschleunigung der Suche in Baumstrukturen durch Monte-Carlo Tree Search[16] ist die Basis für die Erfolge von AlphaZeroGo und AlphaZero[17]. Eine weitere wichtige Grundlage für die schnelle Verarbeitung von großen Datenmengen ist das Locality Sensitive Hashing[1819]. Es reduziert die Komplexität der Berechnung von Approximate Nearest Neighbors[20] und ermöglicht die Nearest Neighbors-Suche in Datenräumen mit tausenden Dimensionen und die Erstellung von Suchindices mit Milliarden Einträgen. So können auch große Datenmengen (Big Data) schnell erfasst und verarbeitet werden.

Die gestiegene Rechenleistung wurde auch genutzt, um komplexere und ausgereiftere Algorithmen für die Erstellung der Datenmodelle zu entwickeln. Unter dem Begriff Deep Learning werden Fortschritte im Bereich von Neuronalen Netzen wie Recurrent Neural Networks und Convolutional Neural Networks beschrieben[21]. Auch was die Robustheit bei der Übertragung der Trainingsergebnisse auf aktuelle Anwendungsszenarien angeht, wurden Fortschritte erzielt. Exemplarisch sei hier die Weiterentwicklung von Decision Trees zu Random Decision Forests genannt. Auf die Auswirkungen des Adversarial Learning kommen wir später zurück.

Last but not least sind in der letzten Zeit Entwicklungs-Frameworks entstanden, die Forschern und Software-Herstellern den einfachen Einsatz von Maschinellem Lernen ermöglichen. Beispielhaft seien hier genannt Googles Tensorflow[22], Amazon Machine Learning[23] und Microsofts Cognitive Toolkit[24] und Azure Machine Learning[25]. [26]

3 Datenarbeit

Manchmal entsteht der Eindruck, dass es mit automatischen maschinellen Verfahren ohne Aufwand möglich ist, gute Ergebnisse zu erzielen. Dieser Eindruck täuscht. Die Ausgangsbasis für den Einsatz von Maschinellem Lernen sind für den Einsatzzweck geeignete Daten in ausreichender Menge. Mit der Qualität der Daten stehen und fallen die erzielten und erzielbaren Resultate. Und die Erhebung von brauchbaren Daten und deren Aufbereitung in eine maschinenverwertbare Form ist zeitraubend und strapaziös. Aber selbst mit penibel aufbereiteten Daten sind die Ergebnisse üblicherweise nicht immer korrekt.

3.1 Lernen und Fehler

Beim Lernen treten Fehler auf. Das gilt auch für ML. Da die meisten Lernverfahren auf Statistik basieren, kann man von zwei klassischen Fehlertypen ausgehen: Fehlalarme (False Positives) und fehlende Erkennung (False Negatives). Wenig Fehlalarme führen zu einem hohen Wert der Kenngröße Precision, während ein hoher Wert der Kenngröße Recall für eine gute Erkennungsleistung steht[27]. Beide Kenngrößen werden im F1 Score zusammengeführt[28].

Bei der Berechnung von Datenmodellen kann die Auswahl und Anzahl der verwendeten Kategorien (bzw. Freiheitsgrade beim Clustering) die Ergebnisse in zwei Richtungen beeinflussen. Entweder sind die extrahierten Datenmodelle zu einfach und können den Anforderungen der Aufgabe nicht gerecht werden. In diesem Fall spricht man von **Underfitting**. Im Gegensatz dazu können beim **Overfitting** die Parameter so detailliert gewählt werden, dass irrelevante Einzelfälle aus dem Datenbestand in die Datenmodelle einfließen. Übergeneralisierte Einzelfälle können dann zu falschen Ergebnissen führen. Nur mit der richtigen Gewichtung der Lernparameter kann das richtige Maß an „Statistical Fit" gefunden werden und sind gute Ergebnisse erzielbar.

3.2 Erkennung verbessern – aber wie?

Maschinelles Lernen kann in vielen Bereichen mit relativ wenig Aufwand ziemlich gute Ergebnisse erzielen. Schwierig wird es allerdings, wenn man die Ergebnisse verbessern möchte. Oft ist es sehr beschwerlich herauszufinden, wie es zu den Gewichtungen und Bewertungen der einzelnen Eingabewerte und letztendlich zur Wahl des jeweiligen Ausgabeergebnisses kommt. Und auch wenn das gelingt, führen manuelle Änderung der internen Bewertung zu schlechteren Ergebnissen an anderer Stelle. Die Systeme haben sich ja in aufwändigen Berechnungen selbst optimiert. Viele Möglichkeiten zur Optimierung bleiben nicht:

15 „A topic-centric list of high-quality open datasets in public domains. By everyone, for everyone" online: https://github.com/awesomedata/awesome-public-datasets

16 Chaslot et al. 2008: Monte-Carlo Tree Search: An New Framework for Game AI. In: Proceedings of the Fourth Artificial Intelligence and Interactive Digital Entertainment Conference. online: https://www.aaai.org/Papers/AIIDE/2008/AIIDE08-036.pdf

17 Deshpande, Nov 2016: Deep Learning Research Review: Reinforcement Learning. Online: https://www.kdnuggets.com/2016/11/deep-learning-research-review-reinforcement-learning.html

18 Broder et al. 1998: „Min-wise independent permutations". Proceedings of the Thirtieth Annual ACM Symposium on Theory of Computing. pp. 327–336

19 Zhao et al. 2014: Locality Preserving Hashing in: Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence S. 2874-2880. online: https://www.aaai.org/ocs/index.php/AAAI/AAAI14/paper/view/8357/8643

20 ANNOY Library. online: https://github.com/spotify/annoy

21 Vgl. Überblick auf https://github.com/terryum/awesome-deep-learning-papers

22 https://www.tensorflow.org/ und https://github.com/tensorflow/tensorflow

23 https://aws.amazon.com/de/aml/

24 https://www.microsoft.com/en-us/cognitive-toolkit/

25 https://studio.azureml.net/

26 Eine umfassende Übersicht gibt: https://www.nodesagency.com/65-frameworks-tools-for-machine-learning/

27 P = Tp / (Tp + Fp); R = Tp / (Tp + Fn) [Tp True positive, Fn False negative] vgl. http://scikit-learn.org/stable/auto_examples/model_selection/plot_precision_recall.html

28 F1 = 2 x (P *R) / P+R

- Die Modelle mit mehr und/oder anderen Daten neu berechnen
- Andere ML-Verfahren anwenden oder erfinden.
- Rechnerische Verfahren zur Parameterreduktion anwenden bzw. andere Verfahren ausprobieren.
- Integration von Domänenwissen und wissenschaftlicher Modelle und Simulationen zur Reduktion von Parametern

Mathematisch versierte Techniker tendieren eher zu den ersten drei Möglichkeiten. Für Experten aus den jeweiligen Fachgebieten ist aber die letzte Variante viel reizvoller. In Bereichen wie z.B. der Medizintechnik oder dem Natural Language Processing hat die Integration von Fachwissen aus relevanten wissenschaftlichen Disziplinen zu besseren und zuverlässigeren Ergebnissen geführt. Während Medizin und Sprachwissenschaft eine jahrtausendealte Tradition als Wissenschaft haben, steht die wissenschaftliche Repräsentation von Cyber-Angriffen noch am Anfang. Ob das zu ähnlichen Erfolgen wie in den etablierten Forschungsfeldern führt, muss sich noch zeigen.

4 Malware automatisch erkennen

Maschinelles Lernen wird auch eingesetzt, um Schadprogrammdateien auf Rechnern zu entdecken. Herkömmliche Signaturen werden durch mathematische Algorithmen ersetzt. Glaubt man den Marketing-Aussagen der Firmen, die maschinell basierte Erkennungsverfahren einsetzen, ist damit das Ende der traditionellen Malware-Erkennung bereits besiegelt. Dabei werden in den Laboren der Hersteller von Virenschutzprogrammen schon seit Jahren Verfahren aus dem Maschinellen Lernen erfolgreich eingesetzt[29]. Dort unterstützen sie die Analysten. Der direkte Einsatz beim Kunden wird wegen der zu hohen Fehleranfälligkeit (s.o.) gescheut.

Es gibt mehrere Ansätze wie Malware-Dateien erkannt werden können: Bei der **statischen Analyse** werden Eigenschaften der Datei und ihres Inhaltes bewertet. Die **dynamische Analyse** nutzt die Aktionen der Malware auf dem infizierten System zur Erkennung. Dabei werden u.a. Informationen über Prozesse, Interprozesskommunikation, Dateisystemzugriffe und Aktionen in der Registry genutzt. Auch die Muster bei der **Netzwerkkommunikation** von Malware können zur Identifikation genutzt werden.

4.1 Schadprogrammdateien erkennen

Für die Erkennung von schädlichen Dateien können sowohl Eigenschaften der Datei (z.B. Dateigröße, Prüfsummen, Dateiname) als auch von deren Inhalten (z.B. Dateistruktur, Entropie) herangezogen werden. Im Prinzip nutzen die ML-Verfahren die gleichen Eigenschaften wie signatur- und regelbasierte Verfahren. Sie haben auch mit ähnlichen Problemen zu kämpfen. Im Laufe der Jahrzehnte haben die Entwickler von Malware viele Wege gefunden, um sich der Entdeckung zu entziehen. Einige der erfolgreichsten und am häufigsten genutzten Verfahren sind:

Code Obfuscation: Die Programmlogik und die Programmstruktur der Malware-Datei kann u.a. durch leere Berechnungen, Aufsplitten von Funktionsblöcken in viele kleine Schnipsel oder komplizierte Reorganisation von Befehlsabfolgen für Analysten schwer lesbar gemacht werden. Bei der maschinellen Verarbeitung werden bekannte Verschleierungsmethoden im Rahmen der Vorverarbeitung normalisiert[30] (z.B. mit REIL[31]).

Anti-Analyse: Im Rahmen des Reverse Engineering der Malware wird versucht deren Schadcode zu rekonstruieren und als Control Flow Graph darzustellen. Die dazu notwendigen Werkzeuge wie Disassembler, Debugger und Emulatoren werden häufig von Malware torpediert.

Laufzeitpacker: ... sind Programme, die den eigentlichen Programmcode komprimieren (ähnlich wie ZIP-Archive) und/oder vor fremden Blicken schützen sollen (z.B. um Spiele vor Raubkopierern zu schützen). Es gibt viele Tausend verschiedene Laufzeitpacker, manche exklusiv in Malware genutzt. Die Meisten werden auch in legaler Software genutzt. Es ist nicht selten, dass Malware mehrere Laufzeitpacker nacheinander nutzt. Die Folge von Laufzeitpackern ist, dass in der äußersten Schicht der Datei nur der Code des Packers sichtbar ist und nicht der eigentlich ausgeführte Code. Die Analyse der Control Flow Graphen dient dadurch allenfalls der Erkennung der äußeren Packerschicht. Diese Hürde müssen auch ML-basierte Erkenner nehmen.

Für die maschinelle Erkennung werden die unterschiedlichsten Merkmale und Inhalte von ausführbaren Dateien herangezogen. Das sind beispielsweise: Anzahl, Name und Größe der Sektionen. Namen der importierten Funktionen aus DLLs, Anzahl der les-, schreib- und ausführbaren Pages, Artefakte von Compiler und Linker, Daten aus der Resource Section (z.B. Programm-Icons, Schriftarten), enthaltene Strings mit Pfaden oder URLs, Packertyp, Entropie der Datei sowie Dateigröße, vorhandene Herstellersignaturen, Datumsangaben. Prinzipiell sind das die gleichen Daten, die auch Malware-Analysten nutzen. Mit einer ausreichend großen Menge an Daten können Maschinen darauf trainiert werden, zwischen gutartiger und bösartiger Software zu unterscheiden.

Die Struktur einer Software zeigt sich im Kontrollflussgraphen (CFG). Er wird als Baumstruktur repräsentiert. Durch einen Vergleich von Abschnitten dieses Baums lassen sich ähnliche Zweige ermitteln. Dadurch ist eine automatisierte Zuordnung von Malware in Familien möglich. Ein analoger Ansatz wurde zur automatischen Bewertung von JavaScripts genutzt. Er basiert auf deren interner Struktur repräsentiert in Form von Abstract Syntax Trees [32]. ALDOCX[33] erkennt schädliche Word-Dokumenten anhand von Strukturmerkmalen.

4.2 An ihren Taten sollst Du sie erkennen

Malware zeichnet sich durch schädliches Verhalten auf bzw. mit dem Computer aus. Diese schädlichen Aktionen können zur Erkennung herangezogen werden. Dabei muss man unterscheiden,

29 Gavrilut, Dragos et al. 2009: [3]

30 Der Vorgang ist vergleichbar mit der Entfernung von doppelten Leerzeichen in einem Text.

31 Dullien & Porst, 2009: REIL: A platform-independent intermediate representation of disassembled code for static code analysis. Online: https://www.researchgate.net/profile/Thomas_Dullien/publication/228958277_REIL_A_platform-independent_intermediate_representation_of_disassembled_code_for_static_code_analysis/links/004635371b8aab56f8000000/REIL-A-platform-independent-intermediate-representation-of-disassembled-code-for-static-code-analysis.pdf

32 Schütt et al., 2012. Early Detection of Malicious Behavior in JavaScript Code. Online: http://ml.informatik.uni-kl.de/publications/earlybird.pdf

33 Nissim et al. März 2017: ALDOCX: Detection of Unkonown Malicious Microsoft Office Documents Using Deignated Active Learning Methods Based on New Structural Feature Extraction Methodology. In: IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY, VOL. 12, NO. 3, S. 631-646

wie die Daten erhoben werden. Virenschutzprogramme mit verhaltensbasierter Erkennung erheben die Daten aus dem laufenden Betrieb. Die meisten Daten dazu werden aber in sog. Sandbox-Systemen erhoben. Sandboxen sind isolierte Systeme, die möglichst viele Aktionen der Prozesse von verdächtigen Programmen und Dateien aufzeichnen. Dazu gehören Aktionen im Dateisystem und in der Registry, Eigenschaften, Aufrufe und Parameter von Systemfunktionen sowie Inhalte der Interprozesskommunikation. Viele Forschungsarbeiten basieren auf Daten, die aus Sandboxen wie z.B. Anubis[34], der CWSandbox[35] oder der Cuckoo Sandbox[36] (die u.a. auch von Avira genutzt wird[37]) erhoben wurden[38]. Um die oft umfangreichen Logfiles der verschiedenen Systeme zu vereinheitlichen und um die Daten für die maschinelle Verarbeitung nutzbar zu machen, müssen die Daten normalisiert werden. Trinius[39] schlägt dazu ein Malware Instruction Set (MIST) vor. Es ist die Basis von Malheur[40], einem Tool zur automatischen Analyse von Malware-Verhalten.

Chumachenko[41] nutzt Aufzeichnungen von 1156 Dateien aus neun verschiedenen Malware-Familien und gutartiger Software in der Cuckoo-Sandbox, und wendet mehrere Lernverfahren auf die Daten an. Die Genauigkeit der Erkennung liegt zwischen 72% und 96%. Sie variiert zwischen den einzelnen Verfahren und Familien. Leider kommt es bei den meisten Ansätzen zu zahlreichen Fehlalarmen im Bereich der gutartigen Software.

4.3 Netzwerkkommunikation

Dieser Bereich kann hier nur angerissen werden. Viele Schadprogramme erzeugen charakteristischen Datenverkehr. Ein Downloader kontaktiert einen Server im Internet, Backdoors und Bots holen sich ihre Instruktionen vom Command&Control Server, Spambots versenden Mails und DDoS-Trojaner schicken viele gleichartige Requests. Der Datenverkehr wird idealerweise von dedizierten Sensoren im Netzwerk erhoben (z.B. Firewall und Intrusion Detection Systeme). Die Daten aus den Logfiles können sehr umfangreich werden, insbesondere wenn sie in SIEM- oder UTM-Systemen mit anderen Daten kombiniert werden. Hier helfen Verfahren zur Anomalieerkennung, die Masse an Daten a) so vorzufiltern, dass sie für menschliche Analysten bearbeitbar werden und b) automatisch ungewöhnliche Vorfälle zu melden und bekannte schädliche Muster zu erkennen.

Bezüglich der Netzwerkdaten kann man zwei Typen unterscheiden: Metadaten der Netzwerkkommunikation und Inhalte der Datenpakete. Wenn man Inhalte der übertragenen Datenpakete bei der maschinellen Bewertung berücksichtigen will, gibt es zwei häufige Hindernisse: a) immer mehr Daten werden verschlüsselt übertragen und b) Privatsphäre und Datenschutz machen eine Überprüfung von Inhalten fragwürdig. Daher werden mittlerweile immer mehr Metadaten der Kommunikation (sog. Flows) für die Auswertung herangezogen. Auch mit diesen Einschränkungen kann Malware erkannt werden[42].

5 Angriffe auf Machine Learning

Machine Learning basiert auf Daten. Das ist auch die Ausgangsbasis für Angriffe auf ML. Während der Trainingsphase versuchen die Algorithmen systematische Strukturen zu finden. Welche Kriterien dabei berücksichtigt werden und mit welcher Gewichtung sie in die Bewertung einfließen ist häufig kaum nachvollziehbar. Daraus kann man folgende Angriffe ableiten:

- Angriffe auf die Erlernbarkeit. Ein Angreifer kann die Daten manipulieren, mit denen die Modelle trainiert werden.
- Angriffe basierend auf den Daten, die bewertet werden. Über die Modelle ist nichts bekannt (Blackbox), aber der Angreifer kennt den Output – entweder genau (z.B. Prozentwert) oder das Urteil (z.B. schädlich vs. harmlos) [43]

Letztlich läuft es darauf hinaus, wer die Kontrolle über die Daten hat, die in das Maschinelle Lernen einfließen. Im ersten Fall zielt das auf die Trainings- und Testdaten ab. Das erfordert spezielle Umstände und gezielte Vorbereitungen auf die hier nicht näher eingegangen werden soll. Im zweiten Fall kontrolliert der Angreifer die zu prüfenden Daten. Diese Möglichkeiten wurden und werden genutzt.

Ein bekanntes Beispiel ist die Erkennung von Spam-Mails per Bayes-Filter[44]. Sie nutzen die Häufigkeit von Wörtern bzw. Zeichenfolgen, um E-Mails als Spam zu klassifizieren. Das funktionierte anfangs sehr zuverlässig[45]. Dann gingen die Spammer aber dazu über a) typische Wörter zu verschleiern oder zu verfälschen und b) ans Ende der E-Mails Abschnitte aus normalen Texten wie z.B. Nachrichten anzuhängen. Dadurch wurden die Kriterien der Bayes-basierten Filter unterlaufen[46].

Bei Malware ist die Lage ähnlich. Die Angreifer können die Dateien nach eigenem Gusto systematisch und automatisiert verändern. Schon bei Bildern wurden sog. Generative Adversarial Networks[47] dazu genutzt, Bilder so zu verfremden, dass z.B. eine Schildkröte als Gewehr erkannt wurde[48]. Analog dazu kann man Dateien so verändern, dass sie nicht mehr als schädlich erkannt werden.

34 Bayer et al, 2006: Dynamic Analysis of Malicious Code.in: Journal in Computer Virology, 2(1):S. 67–77.

35 Willems et al.:März 2007: CWSandbox: Towards Automated Dynamic Binary Analysis. IEEE Security and Privacy, 5(2)

36 Apr 2017: Cuckoo Sandbox Book Release 2.0.0,: Online: https://media.readthedocs.org/pdf/cuckoo/latest/cuckoo.pdf

37 Thorsten Sick, Nov 2014: Cuckoo Sandbox vs. Reality. online: https://blog.avira.com/cuckoo-sandbox-vs-reality-2/

38 Die älteren Sandboxverfahren nutzen zum Aufzeichnen der Daten Agenten auf den (oft virtuellen) Rechnern. Neuere Sandboxen wie etwa VMRay verlegen das Monitoring auf den Hypervisor. Dadurch wird es möglich sehr genaue Aufzeichnungen durchzuführen und viele Tarn- und Ausweichaktionen von Schadprogrammenzu erkennen. Vgl dazu „https://www.vmray.com/resources/" und Analysen auf „https://www.vmray.com/malware-analysis-reports/"

39 Trinius et al., 2009: A Malware Instruction Set for Behavior-Based Analysis. In Technical Report TR-2009-07, University of Mannheim. Online: https://www.sec.cs.tu-bs.de/pubs/2010-sicherheit.pdf

40 Rieck et al. 2011: Automatic Analysis of Malware Behavior using Machine Learning. online: http://www.mlsec.org/malheur/docs/malheur-jcs.pdf

41 Chumachenko, 2017: Machine Learning Methods for Malware Detection and Classification. B.A Thesis. Online: https://www.theseus.fi/bitstream/handle/10024/123412/Thesis_final.pdf

42 Anderson et al., Jul 2016: Deciphering Malware's use of TLS (without Decryption). Online: https://arxiv.org/pdf/1607.01639.pdf

43 Anderson. 2017 [4]

44 Sahami et al., 1998: A Bayesian approach to filtering junk e-mail. In: AAAI'98 Workshop on Learning for Text Categorization.

45 Graham, 2002: A plan for spam. Online: http://www.paulgraham.com/spam.html

46 Vgl. Brodmann, 2005: Evaluierung des Mozilla Spamfilters. Diploma Thesis, TU Darmstadt, S.13. Online: http://www.ke.tu-darmstadt.de/lehre/arbeiten/diplom/2005/Brodmann_Kai.pdf

47 Goodfellow et al, Jun 2014: Generative Adversarial Networks. Online: https://arxiv.org/pdf/1406.2661.pdf

48 Athalye et al, 2017: Synthesizing Robust Adversarial Examples. Online: https://arxiv.org/pdf/1707.07397.pdf

Mit MalGAN[49] wurde Anfang 2017 ein Framework vorgestellt, das systematisch neue Varianten von Schadprogrammen erzeugt. Dabei werden mit einem Adversarial Neuronalen Netz die Eigenschaften verändert, die auch der ML-basierte Malware-Erkenner nutzt (z.B. Aufrufe von System-APIs, Name von Sektionen). Es werden nur Eigenschaften verändert, die nicht kritisch für die Lauffähigkeit der Datei sind. MalGAN prüft, ob die neuen Schaddateien nicht mehr erkannt werden und verbessert so selbständig seine eigenen Testergebnisse. Ähnliche Ansätze wurden auch für die verhaltensbasierte Klassifikation[50] und die automatisierte Erkennung von Android-Malware[51] berichtet.

6 Einsatzgebiete

Die guten Ergebnisse von KI und ML werden auch im Bereich der Erkennung von Malware und Abwehr Hackerangriffen sinnvoll eingesetzt, auch wenn die Ergebnisse in Capture-The-Flag Wettbewerben[52] noch nicht so überzeugend sind wie bei Schach und Go. Andererseits gibt es auch einige konzeptuelle Schwachpunkte. Maschinelles Lernen ist abhängig von gut strukturierten Daten und selbst bei guter Datenlage arbeiten die Verfahren nicht fehlerfrei. Das macht sie für bestimmte Zwecke weniger tauglich als für andere.

6.1 Aktuelle Einsatzgebiete

Ein wichtiger Anwendungsbereich von Maschinellem Lernen besteht im Vorfiltern von großen Datenmengen. Hier kann die Anomalieerkennung im Netzwerk ansonsten unentdeckt gebliebene Aktivitäten offenlegen. Die (Vor-)Klassifikation von verdächtigen Dateiaktivitäten kann die Arbeit der Security-Spezialisten im SOC oder CERT auf ein überschaubares Maß reduzieren und effizienter machen.

Alle etablierten Hersteller von Antiviren-Software setzen in Ihren Laboren Verfahren zur automatischen Klassifikation von Malware ein. Die Verzahnung von Erkennungsverfahren, die auf lokalen Daten basieren mit Informationen zu ähnlichen Vorfällen und Ereignissen, die im Backend der Hersteller vorliegen, wurde in unterschiedlichen Ausprägungen den letzten Jahren bei allen Anbietern von Virenschutzlösungen umgesetzt. Während manche Hersteller beim Marketing den Einsatz von maschinellen Lernverfahren auf den Endpunkten in den Mittelpunkt gestellt haben, verlassen sich andere auf dem Endpunkt auf die bewährten Verfahren und ergänzen sie durch proaktive Erkennungsverfahren und verlagern die maschinelle Verarbeitung ins Backend. Treit[53] zeigt am Beispiel einer Ransomware das Zusammenspiel der einzelnen Verfahren bei Microsoft und wie der lokale Windows Defender mit den verschiedenen Stufen der maschinellen Verarbeitung im Backend in Beziehung steht. Das „Layered detection model“ hat fünf Stufen:

- Lokale Erkennungsverfahren: lokale ML Modelle, Verhaltensbasierte Erkennung, generische und heuristische Signaturen
- Backend-Abfrage von Metadaten: Die maschinelle Analyse der Metadaten erzeugt Regeln im Backend
- Backend. Übertragung der Datei: Die Eigenschaften der Datei werden ermittelt und automatisch klassifiziert.
- Backend. Ausführen der Datei: Die Datei wird in einer Sandbox ausgeführt und die aufgezeichneten Aktionen werden automatisch klassifiziert.
- Backend. Big Data Analytics: Im aufwändigsten Schritt werden die ermittelten Daten mit Expertenregeln und zahlreichen Daten aus anderen Sensoren automatisch abgeglichen.

Im Idealfall können unbekannte Dateien in Sekundenbruchteilen erkannt und Schutzmaßnahmen ausgerollt werden. Gegen die Ransomware aus dem Beispiel waren nach 14 Minuten alle Kunden geschützt.

6.2 Einschränkungen in der Praxis

Die Abhängigkeit von Daten ist eine der größten Einschränkungen für den Einsatz von Maschinellem Lernen. Ohne genügend gute Daten für den Anwendungsfall können keine guten Ergebnisse erzielt werden. Daher ist es wichtig zu wissen mit welchen Daten trainiert wurde und wie gut diese Daten zu meinem Anwendungsszenario passen. ML-ist eher für Standard-Anwendungsfälle geeignet und wird Spezialfälle i.d.R. nicht zuverlässig erkennen können. Das ist gerade im Unternehmensumfeld häufig der Fall. Das Erheben und Aufbereiten von Daten ist mühsam und langwierig. Für den Fall, dass man mit eigenen Daten arbeiten möchte oder muss, sollte man den notwendigen Arbeitsaufwand einkalkulieren.

Der Mangel an Daten hat weitere Folgen. Hersteller sammeln die Daten von Kunden (und Kunden von Kunden) und nutzen sie, um in ihrem Backend die Verfahren und letztlich die Produktqualität zu verbessern. Es gibt auch Szenarien, wo die gesammelten Daten in weiteren Bereichen sehr wertvoll sein können. Das kollidiert regelmäßig mit Anforderungen an die Privatsphäre und den Datenschutz. Insbesondere in Unternehmen und Forschungseinrichtungen ist es geboten, kritische Informationen zu schützen. Es sollte genau geprüft werden, wie der Widerspruch zwischen Datenhunger und Datenschutz umgesetzt ist.

Nicht nur im Bereich der Malware-Erkennung ändern sich Rahmenbedingungen, Vorgehensweisen, Ereignisse uvm. Das führt auch dazu, dass sich die Daten ändern. Es stellt sich die Frage, wie robust die Modelle mit der geänderten Situation umgehen können. Wie stark lässt die Erkennung nach? Wie oft müssen die Modelle neu trainiert werden? Wie aufwändig ist die Update-Prozedur?

Wenn ML-Verfahren, die in der Forschung vielversprechende Ergebnisse gezeigt haben, in die Praxis übertragen werden, muss sich erst zeigen, ob die beschriebenen Verfahren auch mit den tatsächlich auftretenden Datenmengen umgehen können. Viele Testsets für die Erkennung von Schadprogrammen bestehen aus wenigen Familien mit ein paar Tausend Dateien. Um für die tägliche Praxis mit mehreren Tausend Familien und Millionen von Dateien wie sie in CERTs oder bei Herstellern von Virenschutzlösungen vorliegen geeignet zu sein, müssen die Verfahren ska-

49 Hu & Tan, Feb 2017 [5]:

50 Biggio et al, Nov 2014: Poisoning Behavioral Malware Clustering. in Proc. of 7th ACM Workshop on Artificial Intelligence and Security (AISEC),S. 1–10 online: https://www.sec.cs.tu-bs.de/pubs/2014-aisec.pdf

51 Papernot et al. 2017: Practical blackbox attacks against machine learning. In Proceedings oft he 2017 ACM on Asia Conference on Computer and Communications Security. S. 506-519.

52 Devin Coldeway, Aug 2016: Carnegie Mellon's Mayhem AI takes home $2 million from DARPA's Cyber Grand Challenge. Online: https://techcrunch.com/2016/08/05/carnegie-mellons-mayhem-ai-takes-home-2-million-from-darpas-cyber-grand-challenge/

53 Treit, Dez 2017 [6]

lieren. Die dafür notwendigen Anpassungen erfordern viel Zuwendung.

6.3 Der Nutzen für die Wissenschaft

ML macht große Datenmengen für die Lösung bestimmter Aufgaben nutzbar. Der Nutzen für eine Wissenschaft, die nach Modellbildung und Erkenntnisgewinn strebt, ist aber beschränkt. Zu viele Arbeiten geben sich damit zufrieden, dass ML in einer bestimmten Domäne erfolgreich eingesetzt werden kann und dass bestimmte ML-Verfahren bessere Ergebnisse erzielen als andere. Die eigentliche Arbeit an den Daten und deren Bezug zu einer wissenschaftlichen Repräsentation kommen oft zu kurz. Wie lässt sich der Ablauf von Angriffen kategorisieren? Mit welcher Tätergruppe hat man es zu tun? Wie gehen Angreifer taktisch vor? Wie gefährlich ist der aktuelle Angriff? Beziehungen zu solchen, übergeordneten Fragestellungen aus dem Feld der CyberCrime-Forschung wie sie etwa den von MISP unterstützten Taxonomien[54] oder den Klassifikationen von ENISA[55] und MITRE[56] zugrunde liegen, sollten stärker in den datenbezogenen Arbeiten referenziert werden.

7 Zusammenfassung und Fazit

Wir haben gesehen, dass Maschinelles Lernen in der Malware-Erkennung schon seit einiger Zeit erfolgreich eingesetzt wird. Aber es gibt auch Grenzen. Auch ML-Verfahren lassen sich umgehen und sind angreifbar. Durch die ungenaue Erkennung und die zu häufigen Fehler sind sie nicht für alle Anwendungsbereiche geeignet.

Die Basis für die Erkennung sind gute Daten. Für viele Anwendungsbereiche liegen noch nicht genügend brauchbare Daten vor. Die Erhebung von Daten ist aufwändig und kann zu Konflikten mit Privatsphäre und Datenschutz führen.

Maschinelles Lernen ist dort am erfolgreichsten wo die Datenlage konstant ist, und der Anwender die Kontrolle über die Daten hat. Das ist eher im Bereich der Offensive Security als bei der Malware-Abwehr der Fall[57]. Die Angreifer haben volle Kontrolle über die Daten und können automatisiert Dateien erstellen, die von den ML-Verfahren nicht erkannt werden. Die komplexe Dynamik zwischen Angreifer und Verteidiger spiegelt sich in den Daten und überfordert aktuell die Robustheit der abgeleiteten Datenmodelle. In diesem Bereich können mit Adversarial Learning noch große Fortschritte zu erzielt werden. Auch bei der variablen Nutzung von Test- und Trainingssets können kreative Ansätze zu robusteren Datenmodellen führen.

Das kontroverse Verhältnis von massenhafter Datenerhebung zur Verbesserung der Algorithmen und Privatsphäre der Nutzer wird ein ständiger Begleiter bleiben, obwohl die EU-Datenschutzgrundverordnung für eine gewisse Vereinheitlichung sorgen wird. Die dort geforderten Konzepte „Privacy-By-Design" und „Zweckgebundenheit von erhobenen Daten" muss auch umgesetzt werden. Dazu bedarf es schlanker Verfahren, die auch auf Endgeräten mit wenig Rechenleistung funktionieren. Idealerweise werden auch nicht mehr alle Rohdaten übertragen, sondern die daraus abgeleiteten Parameter für das Maschinelle Lernverfahren. So können intelligente Systeme entstehen, die ohne hohe Datentransferraten auskommen und die Anforderungen an die Privatsphäre berücksichtigen. Damit können sich europäische Lösungen von Konkurrenzprodukten aus Fernost und Übersee absetzen.

Die Möglichkeiten von ML und KI werden im Bereich der Malware-Abwehr bereits erfolgreich genutzt. Es ist aber noch viel Raum für Fortschritte. Und die sind auch notwendig, um Angriffe abzuwehren, die mit ML-Unterstützung vorgenommen werden. ML bleibt ein wichtiger und spannender Bereich, auch für die Abwehr von Malware.

54 Vgl. MISP taxonomies online: http://www.misp-project.org/datamodels/#misp-taxonomies

55 Marinos, 2016: ENISA Threat Taxonomy. A tool for structuring threat information. Online: https://www.enisa.europa.eu/topics/threat-risk-management/threats-and-trends/enisa-threat-landscape/etl2015/enisa-threat-taxonomy-a-tool-for-structuring-threat-information

56 ATT&CK, Adversarial Tactics, Techniques & Common Knowledge. online: https://attack.mitre.org/wiki/Main_Page

57 Dullien, 2017 [7]

Literatur

[1] Kubovic, April 2017: Machine Learning und Mathe können keine intelligenten Angreifer überlisten. Online: https://www.welivesecurity.com/deutsch/2017/04/25/machine-learning-kuemmert-angreifer-nicht/

[2] Mayo Mai 2016: Machine Learning Key Terms, Explained. online: https://www.kdnuggets.com/2016/05/machine-learning-key-terms-explained.html und https://www.kdnuggets.com/2016/05/machine-learning-key-terms-explained.html/2

[3] Gavrilut, Dragos et al. 2009: Malware detection using Machine Learning. https://pdfs.semanticscholar.org/1536/bffa0b497ff6cddd65f3cab5f98c6e9bb025.pdf

[4] Anderson. 2017 Blackhat US online: https://www.blackhat.com/docs/us-17/thursday/us-17-Anderson-Bot-Vs-Bot-Evading-Machine-Learning-Malware-Detection-wp.pdf

[5] Hu & Tan, Feb 2017: Generating Adversarial Malware Examples for Black-Box Attacks Based on GAN

[6] Treit, Dez 2017: Detonating a bad rabbit: Windows Defender Antivirus and layered machine learning defenses. online: https://cloudblogs.microsoft.com/microsoftsecure/2017/12/11/detonating-a-bad-rabbit-windows-defender-antivirus-and-layered-machine-learning-defenses/?source=mmpc

[7] Dullien, 2017: Machine learning, offense, and the future of automation. Keynote at ZeroNights 2017. Online: Video: https://www.youtube.com/watch?v=BWFdxAG_TGk , Slides: https://2017.zeronights.org/wp-content/uploads/materials/ZN17_Thomas_Dullien_Machine%20learning,%20offense,%20and%20the%20future%20of%20automation.pdf