Analyse der Anforderungen und Bedrohungslage von komplexen Softwaresystemen

CyberProtect

Baden-Württemberg
MINISTERIUM FÜR WIRTSCHAFT, ARBEIT UND WOHNUNGSBAU

Gefördert von Ministerium für Wirtschaft, Arbeit und Wohnungsbau Baden-Württemberg

Aktenzeichen Zuwendungsbescheid:
3-4332.62-FZI/53

Inhalt

Abbildungsverzeichnis

Analyse der Bedrohungslage von KI-Systemen aus Security Sicht

Im Folgenden werden die im nächsten Kapitel vorgestellten Angriffe aus IT-Security Sicht beschrieben.

Generell ist zu beachten, dass es schwer ist, allgemeine Empfehlungen zur Sicherheit von Systemen zu geben, da die Anforderungen an ein System eine große Rolle bei der Bewertung der Risiken und Auswahl der Schutzmaßnahmen spielen.

Schutzziele

Schutzziele beschreiben Anforderungen an Systeme oder Daten, die erfüllt werden müssen, um den Schutz von Gütern zu gewährleisten.

Sie formalisieren abstrakte Anforderungen und machen diese dadurch einfacher prüf- und messbar.

- **Vertraulichkeit** beschreibt die Eigenschaft, das bestimmte Daten nur von berechtigten Personen oder Systemen eingesehen werden können. Bei KI-Systemen kann es notwendig sein das trainierte System oder die Daten, mit denen es trainiert wurde, geheim zu halten.
- **Integrität** beschreibt den Schutz vor unberechtigter Manipulation von Daten oder Systemen. Hierbei ist zwischen starker- und schwacher Integrität zu unterscheiden. Starke Integrität liegt vor, wenn Daten oder Systeme nicht unbefugt manipuliert werden können, wohingegen schwache Integrität vorliegt, wenn eine unbefugte Manipulation zwar nicht ausgeschlossen werden kann, diese aber einfach erkannt werden kann.

Bei KI-Systemen kann es notwendig sein das System an sich oder die Trainingsdaten, bevor auf ihnen trainiert wird, vor Manipulation zu schützen.

Fähigkeiten von Angreifern

Um die beschriebenen Angriffe auf KI-Systeme zu formalisieren, definieren wir verschiedene Fähigkeiten, über die ein Angreifer verfügen kann.

Nutzung des KI-Systems

Die Nutzung des KI-Systems ist die Möglichkeit, Eingabedaten an das System zu liefern und die Ausgabe des Systems zu erhalten.

Dies kann ein Angreifer z.B. auch über das Internet durchführen, indem er ein öffentlich nutzbares System wie vorgesehen nutzt.

Ggf. muss unterschieden werden, ob die Ausgabe noch gefiltert wird.

So könnte ein Angreifer z.B. bei einem klassifizierenden System nur die Ausgabe mit der höchsten Konfidenz, nicht aber die anderen Ausgaben und die Konfidenzen erhalten.

Es ist zu beachten, dass eine Nutzung des KI-Systems nicht auf dem "Original" erfolgen muss.

Mit gleichem Effekt kann ein Angreifer auch eine identische Kopie des Systems nutzen.

Um die Nutzung des KI-Systems durch den Angreifer zu verhindern, muss daher neben klassischen Maßnahmen zum Zugriffsschutz auch die Vertraulichkeit des Systems gewahrt werden, sodass der Angreifer keine Kopie erstellen kann.

Analyse des KI-Systems

Die Analyse des KI-Systems ist die Möglichkeit des Angreifers, den internen Aufbau eines trainierten Systems zu untersuchen.

Dies beinhaltet z.B. die Struktur eines neuronalen Netzes sowie die erlernten Gewichtungen.

Weiterhin kann ein Angreifer mit der Fähigkeit ein KI-System zu analysieren das System während der Klassifizierung beobachten und erhält Zwischenmodelle und alle Ausgaben.

Diese Fähigkeit ist daher jedenfalls mächtiger als die reine Nutzung des Systems.

Es ist zu beachten, dass eine Nutzung des KI-Systems nicht auf dem "Original" erfolgen muss.

Mit gleichem Effekt kann ein Angreifer auch eine identische Kopie des Systems nutzen.

Um die Nutzung des KI-Systems durch den Angreifer zu verhindern, muss daher neben klassischen Maßnahmen zum Zugriffsschutz auch die Vertraulichkeit des Systems gewahrt werden, sodass der Angreifer keine Kopie erstellen kann.

Lesezugriff auf Trainingsdaten

Bei der Fähigkeit des Angreifers die Trainingsdaten eines Systems zu lesen wird nicht unterschieden, ob er vor- oder nach dem Training Lesezugriff erhält.

Um diese Fähigkeit des Angreifers zu verhindern, muss daher die Vertraulichkeit der Trainingsdaten auch nach Abschluss des Trainings gewahrt werden.

Schreibzugriff auf Trainingsdaten

Schreibzugiff auf die Trainingsdaten ist die Fähigkeit des Angreifers die Trainingsdaten des Systems zu manipulieren.

Dabei kann der Angreifer vor- oder während des Trainings gezielt einzelne oder alle Daten modifizieren, ergänzen oder löschen.

Um dies zu verhindern, muss die Integrität der Trainingsdaten bis zu ihrem tatsächlichen Einsatz gewahrt werden.

Durch einen Schreibzugriff auf die Trainingsdaten während des Trainings ist es einem Angreifer auch möglich ein bereits (teil-) trainiertes System mit weiteren, modifizierten Trainingsdaten zu beeinflussen.

Da eine Manipulation der Trainingsdaten sehr subtil erfolgen kann, kann die schwache Integrität nicht durch menschliche Klassifikatoren verifiziert werden [Shafahi2018].

Es muss sichergestellt werden, dass die Daten auch vor der Klassifikation nicht manipuliert wurden und tatsächlich den zu lernenden Inhalten entsprechen.

Die Integrität der Trainingsdaten muss ab der Quelle gewährleistet werden, wobei zu beachten ist, dass Kameras, Mikrofone oder andere Geräte zur Digitalisierung nicht die Primärquelle sind, da sie nur Abbilder der zu lernenden Objekte erstellen.

Es ist denkbar z.B. einen Hund durch eine bedruckte Klarsichtfolie zu fotografieren oder ein bereits manipuliertes Bild abzufotografieren und damit einen ähnlichen Effekt zu erzielen wie in [Shafahi2018] beschrieben.

Übersicht über Angriffe auf KI-basiere Softwaresysteme

In diesem Anschnitt wird eine Übersicht der Sicherheitsrisiken von KI-basierten Systemen sowie der möglichen Angriffe auf solche Systeme dargestellt. Die Kategorisierung von Attacken basiert auf [Chakraborty2018] sowie [Serban2018].

Abhängig davon, welche Informationen über das System, die Lerndaten sowie den Lernalgorithmus dem Angreifer bekannt sind, wird zwischen den White-Box- und Black-Box-Attacken unterschieden.

Bei White-Box Attacken verfügt der Angreifer über vollständige Kenntnis des Modells (z.B. Art des neuronalen Netzes, Anzahl von Schichten), des Lernalgorithmus (z.B. Gradientenverfahren) sowie hat Zugriff auf Lerndatenverteilung und Parameter des trainierten Modells.

Bei Black-Box Attacken wird dagegen kein Wissen über das Modell vorausgesetzt. Der Angreifer verwendet die Information über die vorherigen Eingaben um die Schwachstellen des Systems zu identifizieren.

Darüber hinaus kann ein Angreifer versuchen entweder die Einsammlung oder die Verarbeitung von Daten zu beeinflussen. Die Angriffszenarien können enstprechend als Vergiftungs- oder Umgehungsattacken kategorisiert werden.

Vergiftungsangriffe (engl. poisoning attacks) finden während der Trainingsphase statt.

Der Angreifer versucht die Lerndaten eines neuronalen Netzes zu kontaminieren indem er die sorgfältig konstruierten Beispiele in die Lerndaten einfügt, dadurch wird der gesamte Lernprozess gefährdet.

Im Falle von Umgehungsangriffen (engl. evasion attacks) versucht der Angreifer das System zu beeinflussen indem er während der Testphase dem Netz böswillige Beispiele präsentiert.

Solche künstlich angefertigten Eingabedaten, die neuronale Netze in die Irre führen, werden gegnerische Eingaben (engl. adversarial inputs) genannt.

Außerdem sind Erkundungsangriffe (engl. exploratory attacks) gegen KI-basierten Systeme möglich.

Hier handelt es sich um die Black-Box-Attacken. Das Ziel des Angreifers ist es, möglichst viel Wissen über den Lernalgorithmus, das zugrundeliegende System sowie Muster in den Lerndaten zu sammeln.

Im Folgenden werden Beispiele für jede Gruppe der Angriffszenarien vorgestellt.

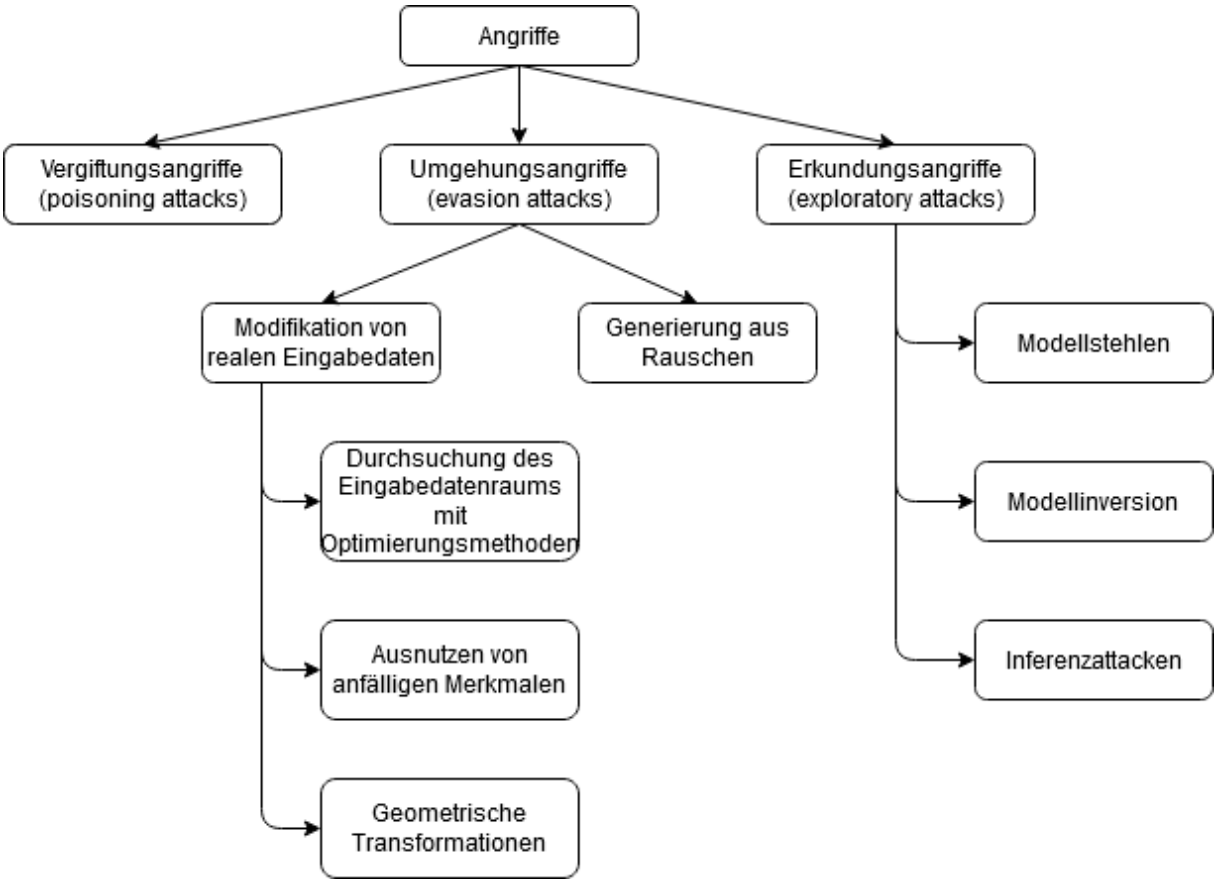

Figure 1 Taxonomie der Angriffe

Vergiftungsangriffe (Poisoning Attacks)

Vergiftungsangriffe sind besonders für Systeme relevant, bei denen Benutzer Feedback verwendet wird um das trainierte Modell anzupassen.

Ein böswilliger Benutzer kann falsches Feedback liefern um das System nach und nach zu vergiften und so das Verhalten des Systems zur Testzeit zu beeinträchtigen.

[Steinhardt2017] beschreibt dass selbst bei Nutzung von effektivsten Verteidigungsmaßnahmen die Korrektklassifikationsrate um 11% sinkt, wenn der Angreifer nur 3% der Lerndaten modifiziert.

In [Muñoz2017] wird ein Back-Gradient Ansatz vorgestellt, dessen Ziel es ist den Gradient durch Differenzierung im Rückwärtsgang zu berechnen, sodass die gesamte Sequenz von Parameter-Updates rekonstruiert werden kann.

[Yang2017] implementiert einen Generator der bösartige Lerndaten zu konstruiert.

Schutzmaßnahmen

Vergiftungsangriffe nutzen manipulierte Trainingsdaten, um Systemen unerwünschtes Verhalten anzutrainieren.

Um dies zu verhindern muss sichergestellt werden, dass keine unautorisierte Instanz Schreibzugriff auf die Trainingsdaten hat oder hatte.

Ein fertig trainiertes Modell im Einsatz ist durch diese Art Angriff nicht mehr gefährdet.

Da Vergiftungsangriffe schon bei einer geringen Anzahl an manipulierten Trainingsdaten einen vergleichsweise großen Effekt haben [Steinhardt2017] können, muss beim Festlegen der Schutzziele für ein konkretes System berücksichtigt werden, ob und wie viele manipulierte Trainingsdaten toleriert werden können.

Ein gezielter Angriff muss nicht alle Trainingsdaten gleichverteilt betreffen, sondern könnte gezielt auf einzelne Kategorien (z.B. ausschließlich Bilder von Hunden) zielen.

Das Generieren von Trainingsdaten aus von Unbekannten erstellten und klassifizierten Inhalten, z.B. von großen Fotoplattformen im Internet erscheint in diesem Kontext gefährlich.

Risiken:

Bias auf gewisse Klassen und Unterdrückung anderer Klassen die in der realen Welt unbiased sind (z.B. face detection mit Bias auf Ethnie)

Gezieltes Einbringen von Anngriffsbeispielen zum gezielten Overfitting mit bestimmtem Efffekt (z.B. Auscschalten von Sicherheitsmechanismen durch bestimmte Muster)

Referenzimplementierungen

- Referenzimplementierung von [Steinhardt2017] - GitHub und Codalab
- Referenzimplementierung von [Yang2017] - GitHub
- Referenzimplementierung von [Shafahi2018] - GitHub

Trojanische Angriffe

Trojanische Angriffe sind eine besondere Art von Vergiftungsangriffen, bei denen das schädliche Verhalten nur durch die Eingaben aktiviert wird, die mit dem Trojaner-Auslöser gestempelt sind. [Liu2017] beschreibt unter anderem trojanische Angriffe auf Gesichtserkennungssystem so dass jedes Gesichtsbild mit dem Auslöser als eine bestimmte Person erkannt wird.

Schutzmaßnahmen

Bei trojanischen Angriffen analysiert ein Angreifer ein bereits trainiertes Netz, um es dann mit manipulierten Daten nachzutrainieren und so unerwünschtes Verhalten einzufügen.

Um dies zu verhindern darf der Angreifer nicht die Fähigkeiten haben das System zu analysieren und die Trainingsdaten zu beschreiben.

Hierbei muss zwischen dem KI-System und Kopien des Systems unterschieden werden, auf welche der Angreifer möglicherweise einfacheren Zugriff hat.

Es ist situationsabhängig zu entscheiden, ob es ausreicht einen Einfluss des Angreifer nur zu erkennen (und ggf. zu revidieren), oder ob dieser komplett unterbunden werden muss (schwache vs. starke Integrität).

Referenzimplementierungen

- Referenzimplementierung von [Liu2017] - GitHub

Umgehungsangriffe (Evasion Attacks)

Seitdem die Anfälligkeit von neuronalen Netzen für adversarial images in [Szegedy2013] entdeckt wurde, wurden mehrere darauf aufbauende Attacken entwickelt.

Diese können nach Art der Erzeugung von gegnerischen Eingaben (durch Modifikation von realen Eingabedaten oder durch Generation aus Rauschen) kategorisiert werden.

Der erste mögliche Ansatz zur Modifikation von Realdaten besteht in Durchsuchung des Raumes von Eigabedaten mithilfe von Optimierungsmethoden.

Weitere Optionen sind das Ausnutzen von anfälligen Merkmalen sowie die Anwendung von geometrischen Transformationen.

Zu den wichtigsten Ansätzen, die auf Optimierung basieren, zählen die L-BFGS (engl. limited memory BFGS) Attacke [Szegedy2013], Deep Fool Attacke [Moosavi2016] sowie Lp Attacke [Carlini2017].

Die zweite Gruppe der Methoden, die die Realdaten anpassen, sucht nach anfälligen Merkmalen in Eingabedaten und modifiziert diese um adversarial images zu erzeugen.

Dazu zählen der FGS (engl. fast gradient sign) Ansatz [Goodfellow2014] sowie JSMA (engl. Jacobian-based saliency map attack) [Papernot2016].

Die letzte Gruppe von modifizierenden Ansätzen nutzt natürliche Perturbationen.

[Engstrom2017] zeigte, dass adversarial images auch mit einfachen Transformationen, und zwar mit Translation und Rotation erzeugt werden können.

In [Xiao2018] werden die böswilligen Bilder durch Änderung der Szenengeometrie erzeugt.

Ein anderer Ansatz beschreibt die Generation von gegnerischen Eingaben aus Rauschen.

Dafür werden folgende generative Modelle eingesetzt: Variational Autoencoders (VAE) und Generative Adversarial Networks (GANs).

Die prominenten Beispiele dafür sind in [Baluja2017] und [Zhao2017] beschrieben.

Alle bisher erwähnten Ansätze können als White-Box-Umgehungsangriffe eingestuft werden.

Die adversarial images, die für ein Modell erzeugt wurden, sind aber oft auf ein anderes Modell übertragbar.

Das ermöglicht auch Black-Box Attacken.

Besonders gefährlich sind aber die Black-Box-Attacken, bei denen gegnerische Eingaben ohne den Einsatz von Zwischenmodellen erzeugt werden und der Angreifer somit nur über die Ausgabedaten des Modells verfügt.

Beispiele für solche Angriffe sind Zoo (engl. zeroth order optimization) [Chen2017] und die Suchmethode in [Narodytska2017]].

Ein Beispiel für einen Umgehungsangriff, bei dem der Detektor getäuscht wird, und so Personen nicht mehr erkennt ist auf Youtube zu finden.

Schutzmaßnahmen

Bei einem White-Box-Umgehungsangriff entwickelt ein Angreifer durch mit verschiedenen Ansätzen optimierten Eingabedaten, Zwischenmodellen und den zugehörigen Ausgaben Adversarial Images.

Um dies zu verhindern, muss verhindert werden, dass der Angreifer die Fähigkeit erhält das KI-System zu analysieren.

Zu beachten ist, dass die aus solchen Angriffen gewonnenen Adversarial Images oft auch auf anderen Modellen übertragbar sind, wenn diese hinreichend ähnlich sind.

Bei einem Black-Box-Umgehungsangriff entwickelt ein Angreifer rein über die Beobachtung der Ausgabe zu von ihm gewählten Eingaben Adversarial Images.

Dies kann nur verhindert werden, in dem die Nutzung des KI-Systems durch den Angreifer unterbunden wird, oder, je nach Form des Angriffs und des KI-Systems, z.B. in der Anzahl begrenzt wird.

Referenzimplementierungen

- Cleverhans - adversarial example library
- foolbox - Python toolbox für Umgehungsangriffe
- Referenzimplementierung von [Carlini2017] - GitHub
- Referenzimplementierung von [Moosavi2016] - GitHub
- Referenzimplementierung von [Chen2017] - GitHub
- Referenzimplementierung von [Zhao2017] - GitHub
- Angriff auf YOLO

Erkundungsangriffe (Exploration Attacks)

Erkundungsangriffe finden während der Testphase statt. Der Angreifer sondiert das System mit gezielt angefertigten Eingabedaten, um Information über das Verhalten des Systems oder über die Lerndaten zu sammeln.

Mögliche Ansätze in dieser Kategorie sind Modellstehlen, Modellinversion sowie Inferenzattacken.

Beim Modellstehlen versucht der Angreifer das Modell zu replizieren.

In [Tramèr2016] wird ein Angriff auf ML-as-a-service demonstriert.

Dabei werden die von dem angegriffenen Modell ausgegebenen Konfidenzen benutzt, um mathematisch die fehlenden Parameter zu bestimmen.

Im Falle von Modellinversion wird angestrebt, die sensiblen Eingabedaten zu extrahieren [Fredrikson2015].

Mitgliedschaftsinferenz verfolgt das Ziel festzustellen, ob die Lerndaten eines angegriffenen Modells eine bestimmte Eingabe beinhalten [Shokri2017].

Security Aspekte

Erkundungsangriffe können von Angreifern zum Extrahieren von Informationen oder geistigen Eigentums aus einem KI-System eingesetzt werden.

Sie sind daher nur gegen Systeme relevant, die vertrauliche Informationen beinhalten oder auf vertraulichen Daten trainiert wurden.

Bei einem Erkundungsangriff zum Stehlen des Modells versucht ein Angreifer über von ihm gewählte Eingabedaten und den vom KI-System zurückgegebenen Ausgabedaten Rückschlüsse auf das verwendete Modell zu ziehen.

Hierbei ist es für den Angreifer hilfreich aber nicht notwendig, wenn das System die Konfidenzen mit ausgibt.

Der Angreifer lernt bei jeder Anfrage an das System etwas über das Modell, der Angriff kann daher nur vollständig verhindert werden, indem die Nutzung des Systems durch den Angreifer unterbunden wird.

Erkundungsangriffe zur Modellinversion oder Mitgliedschaftsinferenz werden mit dem Ziel Informationen über die Trainingsdaten zu lernen eingesetzt.

Hierbei lernt der Angreifer, ob ein bestimmtes Datum Teil der Trainingsdaten war (Mitgliedschaftsinferenz), bzw. extrahiert bestimmte Trainingsdaten aus dem KI-System (Modellinversion).

Da beide Angriffe als Black-Box Angriffe anwendbar sind [Fredrikson2015] [Shokri2017], kann der Angriff nur vollständig verhindert werden, indem die Nutzung des Systems durch den Angreifer unterbunden wird.

Sollte das vollständige Unterbinden der Nutzung des KI-Systems durch den Angreifer nicht möglich sein (z.B. weil der Angreifer nicht eindeutig identifiziert werden kann), muss abhängig vom konkreten KI-System festgelegt werden wie viel der Angreifer über das System lernen darf, und mit welchen Maßnahmen diese Begrenzung durchgesetzt werden kann.

Maßnahmen sind beispielsweise eine Limitierung der Anfragen, die der Angreifer stellen darf und eine Änderung des Systems, sodass Konfidenzen nicht mit ausgegeben werden.

Referenzimplementierungen

- Referenzimplementierung von [Tramèr2016] - GitHub
- Implementierung von [Fredrikson2015] - GitHub
- Referenzimplementierung von [Shokri2017]
  - GitHub
  - Jupyter notebooks
  - Reproduktion von Ergebnissen
- Mia - library for running membership inference attacks against ML models

Angriffe auf Reinforcement Learning

Tiefe neuronale Netze werden in Reinforcement Learning (RL) eingesetzt, um Policies auf rohen Eingaben zu trainieren.

Die Anfälligkeit der Netze für adversarial images kann während des Lernens ausgenutzt werden, um das Verhalten von Policies zu ändern.

Hierfür werden die Ansätze verwendet, die für Umgehungsangriffe eingeführt wurden.

Im Folgenden werden die Angriffe auf RL vorgestellt, die Kategorisierung basiert auf [Chen2019].

Die Anwendungsszenarien für Angriffe auf RL sind Atari Spiele und Pfadplanung.

[Huang2017] beschreibt White-Box- und Black-Box-Angriffe mithilfe von FGS Ansatz.

Die Evaluation erfolgt auf vier Atari Spielen, es wurden drei Ansätze angegriffen: Deep Q Network (DQN), trust policy region optimization (TPRO) und asynchronous advantage actor-critic (A3C).

Dabei wird bei jedem Zeitschritt das von Agenten angesehenes Bild mit FGS manipuliert, sodass das Spiel nicht mehr richtig gespielt wird.

[Kos2017] zeigt dass A3C mit FGS erfolgreich angegriffen wird auch wenn die Störungen nur in einen Teil von Frames eingefügt werden.

Ein White-Box-Angriff auf Pfadplanung mithilfe von RL ist in [Xiang2018] beschrieben.

Hier wird ein Start point-based (SPA) Angriff auf Q-learning vorgeschlagen.

Außerdem wird in [Bai2018] ein Angriff auf DQN-basierte Pfadplanung mithilfe von SPA demonstriert.

[Behzadan2017] demonstriert die Übertragbarkeit von Adversarial-Störungen zwischen verschiedenen DQN Modellen, was Black-Box-Angriffe ermöglicht.

Es wird eine policy induction Attacke (PIA) vorgeschlagen.

Basierend auf der Tatsache, dass das Belohnungssignal gering ist und daher nicht jeder Zeitschritt bei RL angegriffen muss, werden spezifische Zeitschritt-Angriffe in [Lin2017] eingeführt.

Strategically-Timed Attack (STA) formuliert die Auswahl der anfälligen Zeitschritte für den Angriff als Optimierungsproblem.

Eine andere vorgeschlagene Attacke ist Enchanting Attack (EA), dabei wird ein RL Agent forciert, einen erwarteten Zustand zu erreichen.

Auch trojanische Angriffe auf RL Modelle sind möglich.

In [Kiourti2019] wird verstecktes Verhalten in die Funktion von RL Policies eingefügt.

Es wird demonstriert, dass es genügt, nur 0,025% der Lerndaten zu modifizieren um Policies zu erhalten, die sich auf normalen Daten wie gewohnt verhalten, sich aber drastisch verschlechtern, wenn der Trojaner auslöst.

Darüber hinaus können die Angriffe auf RL als one-shot (FGS, SPA) oder iterative (PIA, STA, EA) Ansätze eingegliedert werden.

Referenzimplementierungen

- Referenzimplementierung von [Behzadan2017] - GitHub
- Erkennung von Angriffen auf RL - GitHub

Refrenzen

[Bai2018] Bai, Xiaoxuan, et al. “Adversarial Examples Construction Towards White-Box Q Table Variation in DQN Pathfinding Training.” 2018 IEEE Third International Conference on Data Science in Cyberspace (DSC). IEEE, 2018.

[Baluja2017] Baluja, Shumeet, and Ian Fischer “Adversarial transformation networks: Learning to generate adversarial examples. arXiv preprint

[Behzadan2017] Behzadan, Vahid, and Arslan Munir “Vulnerability of deep reinforcement learning to policy induction attacks.“ International Conference on Machine Learning and Data Mining in Pattern Recognition. Springer, Cham, 2017

[Carlini2017] Carlini, Nicholas, and David Wagner “Towards evaluating the robustness of neural networks“. 2017 IEEE Symposium on Security and Privacy (SP). 2017

[Chakraborty2018] Chakraborty, Anirban, et al. “Adversarial Attacks and Defences: A Survey. “ arXiv preprint

[Chen2017] Chen, Pin-Yu, et al. Zoo: Zeroth order optimization based Black-Box attacks to deep neural networks without training substitute models. Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security. ACM, 2017.

[Chen2019] Chen, Tong, et al. Adversarial attack and defense in reinforcement learning-from AI security view. Cybersecurity 2.1 (2019): 11

[Engstrom2017] Engstrom, Logan, et al. A rotation and a translation suffice: Fooling CNNs with simple transformations. arXiv preprint

[Fredrikson2015] Fredrikson, Matt, Somesh Jha, and Thomas Ristenpart. Model inversion attacks that exploit confidence information and basic countermeasures. Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security. ACM, 2015

[Goodfellow2014] Goodfellow, Ian J., Jonathon Shlens, and Christian Szegedy Explaining and harnessing adversarial examples. International Conference on Learning Representations (ICLR). 2015

[Huang2017] Huang, Sandy, et al. Adversarial attacks on neural network policies. arXiv preprint (2017)

[Kiourti2019] Kiourti, Panagiota, et al. TrojDRL: Trojan Attacks on Deep Reinforcement Learning Agents. arXiv preprint (2019)

[Kos2017] Kos, Jernej, and Dawn Song. Delving into adversarial attacks on deep policies. arXiv preprint (2017)

[Lin2017] Lin, Yen-Chen, et al. Tactics of adversarial attack on deep reinforcement learning agents. arXiv preprint (2017)

[Liu2017] Liu, Yingqi, et al. Trojaning attack on neural networks 2017

[Mirsky2019] Mirsky, Mahler, et al. CT-GAN: Malicious Tampering of 3D Medical Imagery using Deep Learning. arXiv preprint

[Moosavi2016] Moosavi-Dezfooli, Seyed-Mohsen, Alhussein Fawzi, and Pascal Frossard. Deepfool: a simple and accurate method to fool deep neural networks. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2016.

[Muñoz2017] Muñoz-González, Luis, et al. Towards poisoning of deep learning algorithms with back-gradient optimization. Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security. ACM, 2017

[Narodytska2017] Narodytska, Nina, and Shiva Prasad Kasiviswanathan. Simple Black-Box adversarial perturbations for deep networks. IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW). 2017.

[Papernot2016] Papernot, Nicolas, et al. The limitations of deep learning in adversarial settings. 2016 IEEE European Symposium on Security and Privacy (EuroS&P). IEEE, 2016.

[Serban2018] Serban, Alexandru Constantin, and Erik Poll Adversarial Examples-A Complete Characterisation of the Phenomenon. arXiv preprint

[Shafahi2018] Shafahi, Ali, et al. Poison Frogs! Targeted Clean-Label Poisoning Attacks on Neural Networks. Advances in Neural Information Processing Systems 31 (NIPS 2018)

[Shokri2017] Shokri, Reza, et al. Membership inference attacks against machine learning models. 2017 IEEE Symposium on Security and Privacy (SP). IEEE, 2017

[Steinhardt2017] Steinhardt, Jacob, Pang Wei W. Koh, and Percy S. Liang Certified defenses for data poisoning attacks. Advances in neural information processing systems. 2017.

[Szegedy2013] Szegedy, Christian, et al. Intriguing properties of neural networks. International Conference on Learning Representations (ICLR). 2014

[Tramèr2016] Tramèr, Florian, et al. Stealing machine learning models via prediction APIs 25th {USENIX} Security Symposium ({USENIX} Security 16). 2016.

[Xiang2018] Xiang, Yingxiao, et al. A PCA-Based Model to Predict Adversarial Examples on Q-Learning of Path Finding. 2018 IEEE Third International Conference on Data Science in Cyberspace (DSC). IEEE, 2018.

[Xiao2018] Xiao, Chaowei, et al. Spatially transformed adversarial examples arXiv preprint

[Yang2017] Yang, Chaofei, et al. Generative poisoning attack method against neural networks arXiv preprint

[Zhao2017] Zhao, Zhengli, Dheeru Dua, and Sameer Singh. Generating natural adversarial examples. arXiv preprint