Ostbayerische Technische Hochschule Amberg-Weiden
Fakultät Elektrotechnik, Medien und Informatik

Studiengang Elektro- und Informationstechnik

AI Security and Privacy Studienarbeit

von

Fabian Schmidt

Inhaltsverzeichnis

Kapitel 1

Sicherheitskonzept

1.1 Schwachstelle 1: Datenqualität und Label-Verlässlichkeit

Beschreibung:
Die initiale Datensammlung enthält Bilder von fehlerhaften und nicht-fehlerhaften Teilen, die intern erstellt wurden. Nachträglich sollen Bilder, die als fehlerhaft identifiziert wurden, alle 12 Wochen manuell gesichtet und ggf. neu gelabelt werden. Die Qualität und Verlässlichkeit dieser Labels sind kritisch, da ein falsches Label zu einem ineffizienten und ungenauen Modell führen kann.

Konsequenzen:

- Falsche Klassifikation: Ein fehlerhaftes Label könnte dazu führen, dass das Modell falsche Vorhersagen macht, was zu einer Erhöhung der Reklamationskosten und unzufriedenen Kunden führt.
- Vertrauensverlust: Wiederholte Fehlklassifikationen könnte das Vertrauen der Kunden in das Reklamationssystem und die Plattform generell mindern.
- Unbalancierter Datensatz: Da Nutzer nur Bilder defekter Ware hochladen, wird Datensatz mehr und mehr unbalanciert. Falls ein Fehler z.B. Kratzer häufiger vorkommt als Bruchstelle würde das Modell Bruchstellen Fehler seltener erkennen.

Risiko: Hoch

Empfehlungen:

- Datenverifizierung: Implementierung eines doppelt-blinden Bewertungsverfahrens bei der nachträglichen Sichtung der Bilder, um menschliche Fehler zu minimieren.
- Automatisierte Qualitätskontrolle: Einsatz von zusätzlichen Algorithmen zur automatischen Erkennung von Anomalien der Labels.
- Fortlaufende Schulungen: Regelmäßige Schulungen der Mitarbeiter, die die manuelle Nachprüfung durchführen, um deren Genauigkeit und Konsistenz zu

erhöhen.

- Augmentierung unterrepräsentierter Klassen: Vorhandene Bilder sollten augmentiert werden, um sicherzustellen, dass die drei Klassen ausgewogen bleiben.

1.2 Schwachstelle 2: Manipulation der Nutzerdaten

Beschreibung: Kunden können Fotos von Produkten hochladen, um einen Defekt zu melden. Es besteht die Möglichkeit, dass ein Kunde absichtlich manipulierte oder falsche Bilder hochlädt, um eine Ersatzlieferung zu erhalten, obwohl kein echter Defekt vorliegt.

Konsequenzen:

- Betrug: Erhöhte Kosten durch betrügerische Ersatzlieferungen, was zu einem finanziellen Verlust für die Plattform führt.
- Datenverunreinigung: Falsch-positive Daten können in den Trainingssatz gelangen und das Modell verzerren, was die Genauigkeit und Verlässlichkeit der Vorhersagen beeinträchtigt. Solch ein Fehler würde erst nach bis zu 12 Wochen behoben werden.

Risiko: Mittel

Empfehlungen:

- Verifizierungssystem: Einführung eines Verifizierungssystems, bei dem hochgeladene Bilder durch zusätzliche Metadaten wie Seriennummern oder Kaufbelege ergänzt werden müssen.
- Bildanalyse: Nutzung von fortschrittlichen Bildverarbeitungsalgorithmen zur Erkennung von Bildmanipulationen.
- Benutzerverhalten: Analyse des Benutzerverhaltens, um ungewöhnliche Muster und potenziellen Betrug zu identifizieren.

1.3 Schwachstelle 3: Anfälligkeit für Adversarial Attacks

Beschreibung: Da das Modell öffentlich verfügbare Netzarchitekturen und Transfer-Learning nutzt, könnte es anfällig für adversariale Angriffe sein, bei denen speziell manipulierte Bilder verwendet werden, um das Modell zu täuschen.

Konsequenzen:

- Fehlklassifikationen: Adversariale Angriffe können das Modell dazu bringen, Bilder ohne Fehler als defekt zu klassifizieren oder umgekehrt.
- Sicherheitsrisiken: Solche Schwachstellen könnten ausgenutzt werden, um die Plattform systematisch zu schädigen.

Risiko: Hoch

Empfehlungen:

- Robustheitstests: Regelmäßige Durchführung von Robustheitstests, um die Widerstandsfähigkeit des Modells gegen adversariale Angriffe zu überprüfen und zu verbessern.
- Defense-Mechanismen: Implementierung von Defense-Mechanismen wie adversariale Trainingsmethoden, um das Modell gegen solche Angriffe zu härten.
- Monitoring: Ständiges Monitoring der Modellvorhersagen, um ungewöhnliche Muster und potenzielle Angriffe frühzeitig zu erkennen.

Kapitel 2

Studie zur Verteidigung gegen Data Poisoning

Die Implementierung der Studie finden Sie im beiliegenden Jupyter-Notebook. Im Folgenden werden die Ergebnisse der Studie dargelegt. Ziel der Studie ist es, den Effekt von Rauschen als Defense gegen Data Poisoning Angriffe zu untersuchen.

2.1 Trainieren eines Benchmark-Modells

Zu Beginn wurde eine Modell als Benchmark trainiert. Hierzu wurde ResNet18 [3] aus dem timm-Packet [4] verwendet. Das Modell wurde 80 Epochen auf dem CIFAR10 Datensatz [1] trainiert. Das trainierte Modell erreicht eine Test-Genauigkeit von 73.39%.

2.2 Vergiften des Datensatzes

Zur Vergiftung des Datensatzes wird eine Gradient Matching Attacke [2] verwendet, mit $\varepsilon = \frac{16}{255}$ und 1% des Datensatzes wird vergiftet. Ziel ist es, die ersten 10 Bilder der Klasse 0 so zu vergiften, dass sie der Klasse 1 zugeordnet werden. Der Datensatz wird für 500 Epochen vergiftet. Es wird für jedes vergiftete Bild ein Datensatz erstellt, sodass nach Vergiftung 10 Datensätze vorliegen.

Abbildung 2.1: Ergebnis des Data Poisonings

2.3 Trainieren vergifteter Modelle

Beim Training vergifteter Modelle werden die gleichen Parameter verwendet wie in 2.1, um Vergleichbarkeit zu gewährleisten. Es wird jeden der in 2.2 erstellten Datensätze ein Modell trainiert. Danach wird die Effektivität des vergifteten Datensatzes überprüft, indem das vergiftete Bild an ein Modell gegen wird. Die Vergiftung war erfolgreich, wenn das Modell als Klasse 1 klassifiziert. Folgende Tabelle zeigt die Test-Genauigkeit der Modelle und den Erfolg der Vergiftung. Eine 1 in Spalte **Angriff erfolgreich** zeigt, dass der Angriff gegen dieses Modell erfolgreich war.

Modell Nr.	Angriff erfolgreich	Test-Genauigkeit in %
1	0	73.71
2	1	73.02
3	1	72.45
4	1	70.86
5	1	72.49
6	1	71.61
7	1	71.70
8	1	71.43
9	0	73.88
10	1	73.30

Tabelle 2.1: Poison Effektivität

Wie aus der Tabelle entnommen werden kann, weichen die Test-Genauigkeiten der vergifteten Modelle nicht von der Test-Genauigkeit des Benchmark-Modells ab. Im Durchschnitt wurde eine Genauigkeit von 72,45% erreicht. 8 von 10 Modellen wurden erfolgreich vergiftet.

2.4 Rauschen als Verteidigung gegen poisoning

Eine mögliche Verteidigung gegen Poisoning ist das Hinzufügen von Rauschen in den vergifteten Datensatz. Hierzu wird Gauss'sches Rauschen verwendet, mit Mittelwert = 0 und Standardabweichung = 1.

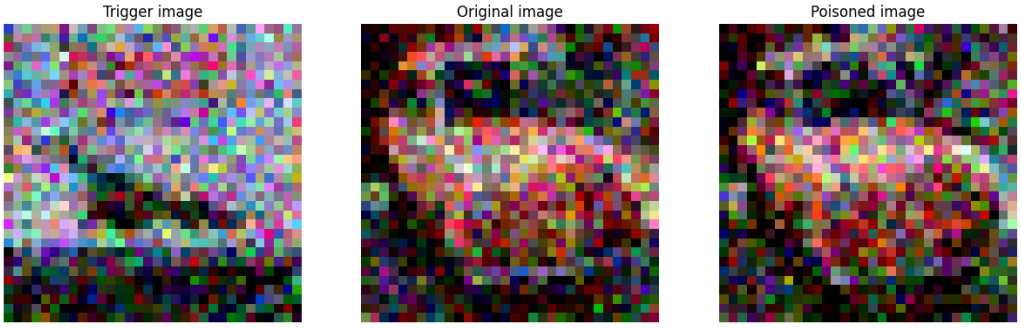

Abbildung 2.2: verrauschter Datensatz

Mit dem verrauschten Datensatz wurden wieder wie in 2.3 10 Modelle trainiert und auf die Effektivität der Vergiftung untersucht.

Modell Nr.	Angriff erfolgreich	Test-Genauigkeit in %
1	0	53.57
2	0	58.70
3	0	56.02
4	0	60.29
5	0	54.67
6	0	59.34
7	0	58.43
8	0	56.06
9	0	55.76
10	0	57.84

Tabelle 2.2: Defense Effektivität

Im Vergleich mit Tabelle 2.1 fällt auf, dass mit der Verteidigung der Angriff gegen keines der Modelle erfolgreich war. Die durchschnittliche Genauigkeit der Modelle liegt bei 57.07%

2.5 Fazit

Die Studie zur Verteidigung gegen Data Poisoning Angriffe zeigt interessante und vielversprechende Ergebnisse. Zunächst wurde ein Benchmark-Modell trainiert, um als Referenz für die Effektivität der Angriffe und Verteidigungsstrategien zu dienen. Das Modell erzielte Test-Genauigkeit von 73,39%.

Durch die Anwendung der Gradient Matching Attacke konnte der Datensatz effektiv vergiften werden. Acht von zehn Modelle wurden erfolgreich manipuliert, was zeigt, dass die Poisoning-Technik in der Lage ist, die Modelle erheblich zu beeinträchtigen, ohne die allgemeine Genauigkeit signifikant zu reduzieren.

Die Implementierung von Rauschen als Verteidigungsmechanismus war erfolgreich. Durch das Hinzufügen von Gauss'schem Rauschen zum vergifteten Datensatz konnte der Angriff in allen Fällen abgewehrt werden. Keines der Modelle wurde erfolgreich vergiftet, was darauf hinweist, dass das Rauschen den Trigger-Effekt der Poisoning-Attacke neutralisiert hat. Allerdings ist anzumerken, dass die Test-Genauigkeit der Modelle unter Einsatz dieser Verteidigungsstrategie auf durchschnittlich 57,07% gesunken ist.

Zusammenfassend lässt sich sagen, dass die Hinzufügung von Rauschen eine effektive Verteidigungsstrategie gegen Data Poisoning darstellt, allerdings zulasten der Modellgenauigkeit. Zukünftige Arbeiten sollten sich darauf konzentrieren, Methoden zu entwickeln, die sowohl die Verteidigung gegen Angriffe sicherstellen als auch die Modellgenauigkeit möglichst hoch halten.

Literatur

[1] *CIFAR-10 and CIFAR-100 datasets*. https://www.cs.toronto.edu/~kriz/cifar.html. (Accessed on 07/12/2024). Juli 2024.

[2] Jonas Geiping u. a. *Witches' Brew: Industrial Scale Data Poisoning via Gradient Matching*. 2021. arXiv: 2009.02276 [cs.CV]. URL: https://arxiv.org/abs/2009.02276.

[3] Kaiming He u. a. *Deep Residual Learning for Image Recognition*. 2015. arXiv: 1512.03385 [cs.CV]. URL: https://arxiv.org/abs/1512.03385.

[4] *timm/resnet18.a1_in1k · Hugging Face*. https://huggingface.co/timm/resnet18.a1_in1k. (Accessed on 07/12/2024). Juli 2024.

Abbildungsverzeichnis

Tabellenverzeichnis