

Projektarbeit

Internationale Hochschule Duales Studium

Studiengang: B.Sc. Informatik

**Inwieweit sind Machine-Learning-Modelle für Netzwerk-Anomalieerkennung zwischen
verschiedenen Datensätzen übertragbar?**

Jonas Weirauch

Matrikelnummer: 10237021

Im Wiesengrund 19, 55286 Sulzheim

Betreuende Person: Dominic Lindner

Abgabedatum: 30.09.2025

Inhaltsverzeichnis

Abbildungsverzeichnis	IV
Abkürzungsverzeichnis	V
1 Einleitung	1
1.1 Motivation und Problemstellung	1
1.2 Forschungsfrage und Zielsetzung	1
1.3 Aufbau der Arbeit	2
2 Theoretische Fundierung	3
2.1 Intrusion Detection Systems und Anomalieerkennung	3
2.2 Detektionsparadigmen: Signatur- versus Anomalieerkennung	3
2.3 Machine Learning-Taxonomie für Anomalieerkennung	3
2.4 Feature Engineering und Datenvorverarbeitung	4
2.5 Transfer Learning und Cross-Dataset-Generalisierung	4
2.6 Evaluationsmetriken	5
3 Methodik	6
3.1 Forschungsdesign und methodische Begründung	6
3.2 Datengrundlage und Stichprobenauswahl	6
3.3 Experimenteller Ablauf und Evaluationsframework	7
3.4 Feature-Engineering und Harmonisierung	7
3.5 Modellauswahl und Hyperparameter-Konfiguration	8
3.6 Evaluationsmetriken und Transfer-Learning-Assessment	8
3.7 Qualitätssicherung und wissenschaftliche Standards	9
4 Ergebnisse	10
4.1 Datensatzinterne Modellperformance	10
4.2 Cross-Validation und Statistische Robustheit	11
4.3 Datensatzübergreifende Transferierbarkeit	11
4.4 Feature-harmonisierte Evaluation	12
4.5 Computational Efficiency	13
5 Diskussion	14
6 Fazit	16
Anhangsverzeichnis	19
A Dataset-Charakterisierung und Explorative Analyse	20
A.1 NSL-KDD Attack Distribution	20
A.2 CIC-IDS-2017 Attack Distribution	21
A.3 Dataset Comparison Overview	22
B Within-Dataset Performance Details	23
B.1 NSL-KDD ROC-Kurven	23

B.2	CIC-IDS-2017 ROC-Kurven	24
B.3	Precision-Recall Kurven	25
B.4	Konfusionsmatrizen NSL-KDD	26
B.5	Konfusionsmatrizen CIC-IDS-2017	26
C	Cross-Validation und Statistische Analysen	27
C.1	Cross-Validation Vergleich	27
C.2	CV Results Distribution	28
C.3	Statistische Vergleichsanalysen	29
C.4	Konvergenzanalyse	30
D	Cross-Dataset Transfer und Generalisierung	31
D.1	Cross-Dataset Transfer Confusion Matrices	31
D.2	Harmonisierte Evaluation	32
E	Learning Curves und Trainingsanalysen	33
E.1	Model Learning Curves	33
F	Computational Efficiency Analysis	35
F.1	Timing Performance Analysis	35
F.2	Real-World Deployment Considerations	36
G	Comprehensive Model Dashboard	37

Abbildungsverzeichnis

1	Vergleichende Modellperformance NSL-KDD vs. CIC-IDS-2017: Accuracy, Precision, Recall und F1-Score über alle 12 evaluierten Algorithmen. Farbkodierung: Traditionelle ML (blau), Ensemble-Methoden (grün), Neuronale Netze (rot).	10
2	Dataset-spezifische Performance-Charakteristika: (a) Accuracy-Scatter NSL-KDD vs. CIC, (b) Metrik-Boxplots, (c) Statistische Signifikanztests ($p < 0.05$).	11
3	Bidirektionale Cross-Dataset-Transfer-Analyse: Performance-Degradation beim Transfer NSL-KDD \leftrightarrow CIC-IDS-2017. Balken zeigen Generalization Gap, Fehlerbalken indizieren Wasserstein Domain Divergence.	12
4	NSL-KDD Attack-Verteilung und Datensatz-Statistiken: (a) Attack-Kategorie-Verteilung (DoS: 36%, Probe: 11%, R2L: <1%, U2R: <1%), (b) Training vs. Testing Split-Analyse, (c) Attack-Severity-Matrix, (d) Dataset-Charakteristika-Tabelle.	20
5	CIC-IDS-2017 Attack-Verteilung und Temporal Patterns: (a) Moderne Attack-Type-Verteilung (14 Kategorien), (b) Temporal Attack Patterns über 5 Tage (3.-7. Juli 2017), (c) Attack-Severity-Heatmap, (d) Vergleichstabelle mit NSL-KDD.	21
6	Vergleichende Dataset-Analyse: (a) Accuracy-Korrelation NSL-KDD vs. CIC (Pearson $r = 0.72$, $p < 0.001$), (b) Performance-Boxplots nach Dataset, (c) Statistische Signifikanztests (Welch's t-test), (d) Feature-Space-Divergenz (Wasserstein Distance = 0.148).	22
7	ROC-Kurven NSL-KDD: (a) Baseline zeigt moderate Trennschärfe (AUC 0.35–1.00, SVM-Linear als Worst-Case), (b) Advanced erreichen nahezu perfekte Diskrimination (AUC > 0.999 für XGBoost, LightGBM, Gradient Boosting). Diagonale = Random Classifier (AUC 0.5).	23
8	ROC-Kurven CIC-IDS-2017: Vergleichbare AUC-Werte wie NSL-KDD, jedoch flacherer Anstieg bei niedrigen FPR-Werten aufgrund höherer Datensatz-Komplexität (79 Features vs. 41, moderne Attack-Vektoren).	24
9	Precision-Recall Trade-Off-Analyse: PR-Kurven sind besonders informativ bei Klassenimbalance (CIC: 83% Normal). Average Precision (AP) aggregiert Performance über alle Schwellenwerte. Baseline-Modelle zeigen stärkeren Precision-Drop bei hohem Recall (rechte Kurvenabschnitte) im Vergleich zu Advanced-Modellen.	25
10	Konfusionsmatrizen NSL-KDD (normalisiert pro True Label): Diagonalelemente = korrekte Klassifikationen (idealer Wert: 1.0). SVM-Linear zeigt starke False-Negative-Rate (dunklere Off-Diagonal-Werte).	26
11	Konfusionsmatrizen CIC-IDS-2017: Naive Bayes zeigt charakteristische Bias zur Attack-Klasse (hohe False-Positive-Rate bei Normal \rightarrow Attack), während Decision Tree nahezu perfekte Klassifikation erreicht (Diagonale ≈ 1.0).	26
12	Cross-Validation Performance-Vergleich NSL-KDD vs. CIC-IDS-2017: 5-Fold stratifizierte CV mit Konfidenzintervallen (95% CI). Fehlerbalken indizieren Variabilität über Folds.	27
13	Boxplot-Verteilung der Cross-Validation Accuracy: Median (zentrale Linie), Interquartilbereich (Box), Whiskers ($1.5 \times \text{IQR}$), Ausreißer (Punkte). SVM-Linear zeigt extreme Variabilität über Folds (IQR = 0.43, Range = 0.33–0.83).	28

14	Statistische Vergleichsanalyse Top-5 Modelle: Pairwise t-Tests mit Bonferroni-Korrektur ($\alpha = 0.01$). Heatmap zeigt p-Werte, Sterne indizieren Signifikanz (** $p < 0.001$, ** $p < 0.01$, * $p < 0.05$).	29
15	Cross-Validation Konvergenzanalyse: Kumulative Mean Accuracy \pm SD über Folds 1–5. Konvergenz ab Fold 3 indiziert ausreichende k-Wahl. Gestrichelte Linie = finale 5-Fold Mean.	30
16	Transfer-Learning Konfusionsmatrizen: (a) NSL-KDD \rightarrow CIC-IDS-2017, (b) CIC-IDS-2017 \rightarrow NSL-KDD für XGBoost. Forward-Transfer (a) zeigt moderate Generalisierung (Target Acc = 0.827), Reverse-Transfer (b) zeigt starke Degradation (Target Acc = 0.431).	31
17	Harmonisierte Cross-Dataset Evaluation: Performance bei PCA-alignierten Features (20 Komponenten, 94.7% erklärte Varianz). Threshold-Tuning via Grid Search (0.1–0.9 in 0.1-Schritten).	32
18	Lernkurven Top-3 Modelle bei variierenden Trainingsdatengrößen (1k–100k Samples): Training Accuracy (durchgezogene Linie) vs. Validation Accuracy (gestrichelt). Schattierte Bereiche = 95% CI über 3 Wiederholungen.	33
19	Training Time vs. Accuracy Trade-Off: Bubble-Chart mit Bubble-Größe proportional zu Inferenzzeit. Optimale Modelle in oberer linker Region (hohe Accuracy, niedrige Training Time).	35
20	Comprehensive Multi-Metrik Dashboard: (a) Radar-Chart aller Performance-Metriken, (b) Parallel-Koordinaten-Plot für Metrik-Interaktion, (c) Hierarchische Clustering-Dendrogramm ähnlicher Modelle, (d) Principal Component Biplot für Modell-Distanzen im Metrik-Raum.	37

Abkürzungsverzeichnis

AI	Artificial Intelligence
AUC	Area Under the Curve
CIC-IDS-2017	Canadian Institute for Cybersecurity Intrusion Detection System 2017
DoS	Denial-of-Service
EFB	Exclusive Feature Bundling
FPR	False Positive Rate
GOSS	Gradient-based One-Side Sampling
HIDS	Host-based Intrusion Detection Systems
IDS	Intrusion Detection Systems
k-NN	k-Nearest Neighbors
ML	Machine Learning
MLP	Multi-Layer Perceptron
NIDS	Network-based Intrusion Detection Systems
NSL-KDD	Network Security Laboratory - Knowledge Discovery and Data Mining
ROC	Receiver Operating Characteristic
SVM	Support Vector Machines
TPR	True Positive Rate

1 Einleitung

1.1 Motivation und Problemstellung

Mit über 10,5 Billionen US-Dollar geschätzten jährlichen Schäden bis 2025 stellen Cyberangriffe eine der größten globalen Bedrohungen dar (World Economic Forum, 2024). Diese besorgniserregenden Statistiken unterstreichen die akute Notwendigkeit wirksamer Sicherheitsvorkehrungen zum Schutz kritischer Infrastrukturen (Taman, 2024).

Traditionelle signaturbasierte Intrusion Detection Systeme (IDS) erreichen zunehmend ihre Grenzen bei der Erkennung neuartiger Zero-Day-Exploits und unbekannter Angriffsmuster (Belavagi & Muniyal, 2016; Ring et al., 2019). Machine Learning (ML) bietet das Potenzial, diese Limitationen zu überwinden, jedoch ist die tatsächliche Wirksamkeit verschiedener ML-Modelle in heterogenen Netzwerken noch nicht vollständig geklärt. Ein kritisches Problem stellt dabei die Generalisierungsfähigkeit dar: Während Modelle auf spezifischen Trainingsdaten exzellente Leistungen erzielen, zeigen sie oft dramatische Leistungseinbußen beim Transfer auf neue Netzwerkumgebungen (Ring et al., 2019).

Forschungslücke: Bisherige Studien konzentrieren sich primär auf Within-Dataset-Evaluationen und vernachlässigen die praktisch relevante Frage der Cross-Dataset-Transferierbarkeit (Mourouzis & Avgousti, 2021). Die systematische Bewertung der Generalisierungsfähigkeit zwischen fundamental verschiedenen Netzwerk-Datensätzen, insbesondere zwischen historischen Benchmarks wie NSL-KDD und modernen Datensätzen wie CIC-IDS-2017, bleibt eine unzureichend erforschte, aber praxiskritische Herausforderung.

1.2 Forschungsfrage und Zielsetzung

Diese Arbeit untersucht systematisch die Generalisierungsfähigkeit von zwölf ML-Modellen über zwei fundamental unterschiedliche Netzwerk-Datensätze hinweg. Die zentrale Forschungsfrage lautet:

„Inwieweit sind Machine-Learning-Modelle für Netzwerk-Anomalieerkennung zwischen verschiedenen Datensätzen übertragbar?“

Die Untersuchung fokussiert sich auf die Cross-Dataset-Transferierbarkeit zwischen NSL-KDD (1998, 41 Features) und CIC-IDS-2017 (2017, 79 Features), die sich fundamental in Datenverteilung, Merkmalsdimensionalität und Angriffsszenarien unterscheiden (Mourouzis & Avgousti, 2021).

Die konkreten Forschungsziele umfassen erstens die **systematische Cross-Dataset-Evaluation** durch bidirektionale Transferanalyse mit zwölf ML-Algorithmen von Baseline-Modellen (Random Forest, Decision Tree, k-NN, Logistic Regression, Naive Bayes, Linear SVM) bis zu Advanced-Modellen (XGBoost, LightGBM, Gradient Boosting, Extra Trees, MLP, Voting Classifier). Zweitens erfolgt die **Entwicklung neuartiger Transfer-Metriken** durch Einführung von Generalization Gap, Transfer Ratio und Relative Performance Drop als quantitative Maße für Cross-Dataset-Robustheit. Drittens wird eine **Feature-Space-Harmonisierung** zur Überbrückung der Dimensionalitätslücke (41 vs. 79 Dimensionen) implementiert. Viertens zielt die Arbeit auf **praktische Deployment-Guidance** durch Identifikation der transferrobustesten Algorithmen für heterogene Netzwerkumgebungen ab.

1.3 Aufbau der Arbeit

Die Arbeit gliedert sich in vier aufeinander aufbauende Hauptteile. Zunächst werden in den *theoretischen Grundlagen* die konzeptionellen Fundamente der Netzwerk-Anomalieerkennung etabliert, einschließlich einer Taxonomie der eingesetzten Machine-Learning-Verfahren (McHugh, 2000; Vinayakumar et al., 2019).

Im *methodischen Teil* wird das dreistufige Evaluationsframework vorgestellt, das Within-Dataset-Validation, Cross-Dataset-Transfer und Feature-Harmonisierung systematisch kombiniert (Gharib et al., 2016).

Die *empirische Analyse* präsentiert die Ergebnisse der umfassenden Modellvergleiche zwischen NSL-KDD und CIC-IDS-2017. Neben klassischen Performance-Metriken werden neuartige Transfer-Kennzahlen wie Generalization Gap und Transfer Ratio eingeführt.

Abschließend werden in der *Diskussion* die praktischen Implikationen für IDS-Deployments erörtert. Der wissenschaftliche Beitrag liegt in der erstmaligen systematischen Cross-Dataset-Evaluation von zwölf ML-Modellen unter realistischen Transferbedingungen sowie der Entwicklung neuartiger Transfer-Metriken für ML-basierte Cybersecurity-Systeme.

2 Theoretische Fundierung

2.1 Intrusion Detection Systems und Anomalieerkennung

Intrusion Detection Systems (IDS) fungieren als automatisierte Frühwarnsysteme zur Identifikation ungewöhnlicher Netzwerkmuster, die auf Sicherheitsbedrohungen hindeuten (Ring et al., 2019; Vinayakumar et al., 2019). Die **architektonische Klassifikation** erfolgt nach Einsatzort: **Network-based IDS (NIDS)** überwachen zentral den Netzwerkverkehr, während **Host-based IDS (HIDS)** lokale Systemaktivitäten analysieren (Gharib et al., 2016). **Deployment-Modi** unterscheiden zwischen passiver Überwachung (latenz-neutral) und aktiver Inline-Filterung (proaktive Mitigation bei Durchsatz-Limitationen).

Anomalien lassen sich in drei Kategorien differenzieren (Ring et al., 2019): **Punktuelle Anomalien** (einzelne Abweichungen von der Normalverteilung), **kontextuelle Anomalien** (situationsabhängige Unregelmäßigkeiten) und **kollektive Anomalien** (koordinierte Gruppenmuster wie Botnet-Aktivitäten). Moderne Netzwerkkumgebungen sind durch **Concept Drift** charakterisiert – zeitliche Veränderungen der statistischen Datenverteilung, die die Generalisierungsfähigkeit von Detektionssystemen herausfordern (Gharib et al., 2016).

2.2 Detektionsparadigmen: Signatur- versus Anomalieerkennung

Die Anomalieerkennungstechnologien differenzieren sich in zwei fundamentale Paradigmen (Belavagi & Muniyal, 2016; Ring et al., 2019): **Signaturbasierte Systeme** vergleichen Netzwerkverkehr mit Datenbanken bekannter Angriffsmuster und erzielen hohe Präzision bei katalogisierten Bedrohungen, versagen jedoch bei Zero-Day-Exploits und polymorphen Angriffen (Vinayakumar et al., 2019). **Anomaliebasierte Systeme** etablieren statistische Modelle normalen Netzwerkverhaltens und klassifizieren Abweichungen als potenzielle Bedrohungen, wodurch unbekannte Angriffsmuster detektiert werden können, jedoch bei höheren False-Positive-Raten.

Machine Learning-Ansätze überwinden diese Limitationen durch überwachte Verfahren, die komplexe nichtlineare Feature-Beziehungen erlernen, und unüberwachte Methoden zur Detektion neuartiger Anomaliemuster (Vinayakumar et al., 2019). Deep Learning ermöglicht automatische Feature-Extraction aus hochdimensionalen Netzwerkdaten und eliminiert manuell entwickelte Heuristiken (Goodfellow et al., 2016).

2.3 Machine Learning-Taxonomie für Anomalieerkennung

Die Evaluation implementiert zwölf ML-Algorithmen in zwei Kategorien (Vinayakumar et al., 2019): **Baseline-Modelle** (Random Forest, Decision Tree, Logistic Regression, Naive Bayes, k-NN, Linear SVM) repräsentieren etablierte, interpretierbare Verfahren mit geringen computational Anforderungen. **Advanced-Modelle** (XGBoost, LightGBM, Gradient Boosting, Extra Trees, MLP, Voting Classifier) umfassen moderne Algorithmen mit erhöhter Komplexität und superior Generalisierungsfähigkeit.

Baseline-Charakteristika: Random Forest nutzt Bootstrap Aggregating zur Overfitting-Reduktion (Hastie et al., 2009). Logistic Regression modelliert Klassenwahrscheinlichkeiten via Sigmoid-Funktion $P(y = 1|x) = \frac{1}{1+e^{-(\beta_0+\beta_1 x)}}$ (Bishop, 2006). Naive Bayes basiert auf dem Bayes'schen Theorem

mit Unabhängigkeitsannahme $P(x_1, \dots, x_n|y) = \prod_{i=1}^n P(x_i|y)$. k-NN implementiert instanzbasiertes Lernen, ist jedoch anfällig für die Curse of Dimensionality (Hastie et al., 2009). Linear SVM optimiert die Hyperebene $w^T x + b = 0$ zur Margin-Maximierung (Platt, 1999).

Advanced-Charakteristika: XGBoost implementiert optimiertes Gradient Boosting mit Regularisierung $\mathcal{L} = \sum_{i=1}^n l(y_i, \hat{y}_i) + \sum_{k=1}^K \Omega(f_k)$ (Hastie et al., 2009). LightGBM nutzt Gradient-based One-Side Sampling (GOSS) und Exclusive Feature Bundling (EFB) zur Effizienzsteigerung (Zhou et al., 2020). MLP implementiert universelle Funktionsapproximation durch neuronale Architekturen (Goodfellow et al., 2016). Voting Classifier aggregiert heterogene Lerner: $\hat{y} = \arg \max_c \sum_{i=1}^m w_i \cdot P_i(c|x)$ (Hastie et al., 2009).

2.4 Feature Engineering und Datenvorverarbeitung

NSL-KDD umfasst 41 Features (kategoriale und numerische Verbindungsattribute), während **CIC-IDS-2017** 79 Flow-basierte Features bereitstellt (Gharib et al., 2016; Sharafaldin et al., 2018). Distanzbasierte Algorithmen (k-NN, SVM) und neuronale Netze erfordern Skalierung: Min-Max $x_{norm} = \frac{x - x_{min}}{x_{max} - x_{min}}$ oder Z-Score $x_{std} = \frac{x - \mu}{\sigma}$ (Bishop, 2006). Tree-basierte Modelle sind skalierungsinvariant.

Klassenimbalance (normale Verbindungen: 95-99%) wird durch Class Weight Balancing adressiert: $w_c = \frac{n_{samples}}{n_{classes} \cdot n_{samples}_c}$ (Hastie et al., 2009; Ring et al., 2019).

2.5 Transfer Learning und Cross-Dataset-Generalisierung

Transfer Learning beschreibt die Fähigkeit, Wissen aus einer Quelldomäne zur Performance-Verbesserung in einer Zieldomäne zu nutzen (Goodfellow et al., 2016). Netzwerk-Datensätze differieren fundamental in Feature-Dimensionalität (NSL-KDD: 41 vs. CIC-IDS-2017: 79), temporaler Abdeckung (historische vs. moderne Angriffsmuster) und Topologie (simuliert vs. real), was zu **Distribution Shift** der Joint-Probability-Distribution $P(X,Y)$ führt.

Concept Drift charakterisiert zeitliche Datenverteilungsänderungen, kritisch für evolvierende Cyberbedrohungen (Ring et al., 2019). Die Varianten umfassen **Covariate Shift** (Änderung von $P(X)$ bei konstantem $P(Y|X)$), **Prior Probability Shift** (Änderung der Klassenverteilung $P(Y)$) und **Concept Shift** (fundamentale Änderung von $P(Y|X)$).

Die **Cross-Dataset-Robustheit** wird durch neuartige Transfer-Metriken quantifiziert:

$$\text{Generalization Gap} = \text{Performance}_{\text{source}} - \text{Performance}_{\text{target}} \quad (1)$$

$$\text{Transfer Ratio} = \frac{\text{Performance}_{\text{cross-dataset}}}{\text{Performance}_{\text{within-dataset}}} \quad (2)$$

Werte nahe 1.0 indizieren hohe Transferierbarkeit, während niedrige Werte domänenspezifische Überanpassung signalisieren.

2.6 Evaluationsmetriken

IDS-spezifische Metriken berücksichtigen Klassenimbalance und Deployment-Anforderungen (Belavagi & Muniyal, 2016). **Accuracy** ist bei imbalancierten Datensätzen irreführend (always normal-Klassifikator: 95% Accuracy). **F1-Score** harmonisiert Precision und Recall: $F_1 = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall}$ (Hastie et al., 2009; Mourouzis & Avgousti, 2021). **5-fache Kreuzvalidierung** mit Stratifizierung verhindert Datenleckage (Tavallaee et al., 2009).

3 Methodik

3.1 Forschungsdesign und methodische Begründung

Die vorliegende Arbeit verfolgt ein **dreistufiges quantitatives Evaluationsframework**, das systematisch von datensatzinterner Validierung über bidirektionale datensatzübergreifende Transfers bis zu merkmalsharmonisiertem Transfer fortschreitet. Diese Komplexitätssteigerung ermöglicht eine vollständige Charakterisierung der Transferierbarkeit von Machine-Learning-Modellen zwischen historischen (NSL-KDD, 2009) und modernen (CIC-IDS-2017, 2017) Netzwerkkumgebungen.

Die **Wahl eines quantitativen Designs** begründet sich in der Notwendigkeit, objektive Leistungsunterschiede zwischen zwölf ML-Algorithmen unter kontrollierten Bedingungen zu quantifizieren. Die Forschungsfrage erfordert messbare Generalisierungsmetriken, die qualitative Ansätze nicht liefern können. Die vollständig automatisierte Experimentalpipeline gewährleistet verzerrungsfreie Evaluation mit deterministischen Ergebnissen (RANDOM_STATE=42).

Das Forschungsdesign umfasst drei hierarchische Evaluationsebenen. Zunächst erfolgt eine **datensatzinterne Validierung** mittels 5-fach stratifizierter Kreuzvalidierung, die Referenzwerte für die spätere Transferbewertung schafft (Tavallaee et al., 2009). Im nächsten Schritt folgt die **datensatzübergreifende Transferanalyse**, bei der Modelle, die auf NSL-KDD trainiert wurden, auf CIC-IDS-2017 getestet werden und umgekehrt. Diese bidirektionale Evaluation deckt potenzielle Asymmetrien auf (Ring et al., 2019). Abschließend wird im **Featureharmonisiertem Transfer** die Inkompatibilität der Feature-Räume (41 vs. 79 Dimensionen) durch PCA-basierte Ausrichtung adressiert, bei dem beide Datensätze auf einen gemeinsamen 20-dimensionalen latenten Raum projiziert werden und dabei über 95 Prozent der Varianz erhalten bleiben (Goodfellow et al., 2016).

Die methodische **Innovation** liegt in der Quantifizierung mittels neuartiger Transfer-Metriken: **Transfer Ratio** ($TR = \text{Performance}_{\text{target}} / \text{Performance}_{\text{source}}$), **Generalization Gap** ($GG = \text{Performance}_{\text{source}} - \text{Performance}_{\text{target}}$) und **Relative Performance Drop** ($RPD = (GG / \text{Performance}_{\text{source}}) \times 100$) (Mourouzis & Avgousti, 2021).

3.2 Datengrundlage und Stichprobenauswahl

Die empirische Evaluation basiert auf zwei etablierten Benchmark-Datensätzen. **NSL-KDD (2009)** stellt eine kuratierte Revision des KDD Cup 99-Datensatzes dar und umfasst 125.973 Trainingsproben sowie 22.544 Testproben mit vier Angriffskategorien (DoS: 36%, Probe: 11%, R2L/U2R: <1%) (Tavallaee et al., 2009). Der Feature-Space besteht aus 41 verbindungsbasierten Attributen wie Protokoll-Typ, Verbindungsdauer und übertragene Bytes. Die Daten basieren auf simulierten Netzwerkangriffen aus 1998 (McHugh, 2000).

CIC-IDS-2017 (2017) repräsentiert eine moderne Alternative mit 2.830.540 Proben aus realistischem Netzwerkverkehr über fünf Tage (Sharafaldin et al., 2018). Der Datensatz umfasst 14 moderne Angriffskategorien (Heartbleed, SQL-Injection, DDoS-Varianten) und 79 bidirektionale Flow-Features mit erweiterten statistischen Charakterisierungen. Die Klassenverteilung (83% Normal, 17% Angriff) ist realistischer als bei NSL-KDD.

Die **Stichprobenauswahl** folgt einem speicheradaptiven Protokoll. Bei Systemen mit über 16 GB

Arbeitsspeicher wird der vollständige Datensatz verwendet, während bei geringeren Ressourcen eine stratifizierte Zufallsstichprobe gezogen wird, die die ursprüngliche Klassenverteilung proportional erhält. Für CIC-IDS-2017 erfolgt bei Speicherbeschränkungen eine Reduktion auf 200.000 bis 500.000 Proben. Beide Datensätze wurden im März 2025 heruntergeladen und mittels SHA-256-Prüfsummen validiert. Die Ergebnisse aus dieser Arbeit stammen aus den vollständigen Datensätzen.

3.3 Experimenteller Ablauf und Evaluationsframework

Das experimentelle Framework implementiert eine vollständig automatisierte 8-stufige Pipeline, die als eigenständige Python-Skripts konzipiert ist. Die Pipeline beginnt mit explorativer Datenanalyse zur Validierung der Datensatzladung und Identifikation von Datenqualitätsproblemen. Anschließend erfolgt das Training von sechs Baseline-Algorithmen (Random Forest, Logistic Regression, Decision Tree, Naive Bayes, k-NN, Linear SVM) und sechs Advanced-Modellen (XGBoost, LightGBM, Gradient Boosting, Extra Trees, MLP, Voting Classifier). Zur Adressierung der Klassenimbalance wird systematisch `class_weight="balanced"` verwendet (Hastie et al., 2009; Vinayakumar et al., 2019).

Die Within-Dataset-Robustheit wird durch 5-Fold Stratified Cross-Validation evaluiert, bei der die Aggregation mittels Mittelwert und Standardabweichung erfolgt und durch Bootstrap-Konfidenzintervalle (1000 Iterationen, 95% CI) ergänzt wird. Paired t-Tests mit Bonferroni-Korrektur identifizieren signifikante Performance-Differenzen zwischen Modellen. Das Kernexperiment implementiert bidirektionale Transfer-Tests, bei denen Modelle ohne Retraining auf dem Target-Dataset evaluiert werden. Die Wasserstein-Distanz zwischen Feature-Distributionen quantifiziert die theoretische Domain-Divergenz.

Für das Feature-Space-Alignment werden beide Datensätze mittels StandardScaler auf Basis der Source-Statistiken skaliert und auf 20 Hauptkomponenten projiziert. Für CIC-IDS-2017 wird Incremental Learning via SGDClassifier mit `partial_fit()` in 20.000-Sample-Batches implementiert (Bishop, 2006). Die Pipeline konsolidiert alle Ergebnisse in CSV- und JSON-Dateien und generiert abschließend publikationsreife Visualisierungen (300 DPI, PDF-Export) mit colorblind-friendly Paletten.

Die technische Infrastruktur basiert auf Python 3.8+ mit scikit-learn 1.3+, XGBoost 1.7+ und LightGBM 3.3+. Die Pipeline inkludiert automatische Umgebungsvalidierung und gewährleistet Fault-Tolerance durch inkrementelle Ergebnisspeicherung.

3.4 Feature-Engineering und Harmonisierung

Die Inkompatibilität zwischen NSL-KDD (41 Features) und CIC-IDS-2017 (79 Features) erfordert eine **Zwei-Ebenen-Harmonisierungsstrategie**. Auf der ersten Ebene erfolgen semantische Feature-Mappings, bei denen durch Domain-Knowledge sechs gemeinsame Features identifiziert wurden: `duration`, `forward_bytes`, `backward_bytes`, `total_bytes`, `bytes_per_second` und `byte_ratio`. Diese Features sind protokollunabhängig, korrelieren mit Angriffsindikatoren und liegen in beiden Datensätzen mit vergleichbarer semantischer Definition vor (Gharib et al., 2016). Die Validierung erfolgte mittels Korrelationsanalyse (Pearson $r > 0,85$).

Auf der zweiten Ebene wird eine statistische Ausrichtungspipeline implementiert. Die StandardScaler-Normalisierung erfolgt als Z-Score-Transformation unter Verwendung der Quelldatensatz-Statistiken, sodass der Zieldatensatz durch die "Linse" des Quellmodells betrachtet wird (Goodfellow et al., 2016). Anschließend werden beide Datensätze auf die ersten 20 Hauptkomponenten projiziert, die über

95 Prozent der kumulativen Varianz erklären. Dieser Kompromiss wurde empirisch validiert: 15 Komponenten führten zu höherer Transferdegradation, während 25 Komponenten nur marginale Verbesserungen bei erhöhter Overfitting-Gefahr zeigten.

Die Domänendivergenz-Quantifizierung erfolgt mittels Wasserstein-Distanz zwischen den PCA-angepassten Feature-Verteilungen. Die Implementierung inkludiert automatisierte Fehlerbehandlung mit UTF-8-Parsing, fehlende Werte-Imputation und IQR-basierter Ausreißer-Erkennung, wobei alle Vorverarbeitungsschritte ausschließlich auf Basis des Quelldatensatzes parametrisiert werden, um Datenleckage zu vermeiden.

3.5 Modellauswahl und Hyperparameter-Konfiguration

Die Evaluation umfasst zwölf Algorithmen, die sich in Baseline- und Advanced-Modelle unterteilen. Die sechs Baseline-Modelle umfassen Random Forest ($n_estimators=200$, $max_depth=25$), Logistic Regression ($solver=\text{lbfgs}$, $C=1.0$), Decision Tree ($max_depth=25$), k-Nearest Neighbors ($k=3$), Linear SVM ($kernel=\text{linear}$) und Naive Bayes (GaussianNB). Diese decken Ensemble-Methoden, lineare Klassifikatoren, instanzbasiertes und probabilistisches Lernen ab (Hastie et al., 2009).

Die sechs Advanced-Modelle umfassen XGBoost ($n_estimators=400$, $max_depth=6$, $learning_rate=0.1$), LightGBM ($n_estimators=400$, $learning_rate=0.05$, $num_leaves=31$), Gradient Boosting, Extra Trees, ein Multi-Layer Perceptron ($hidden_layers=(128,64)$, ReLU-Aktivierung, vorzeitiges Stoppen) und einen Voting Classifier mit Soft-Voting-Kombination. Diese repräsentieren moderne ML-Praxis für Cybersecurity-Anwendungen (Vinayakumar et al., 2019). Die Hyperparameter folgen praxisorientierten Standardkonfigurationen ohne umfassende Rastersuche, um Hyperparameter-Überanpassung zu vermeiden. Die Behandlung des Klassenungleichgewichts erfolgt durch `class_weight="balanced"`.

3.6 Evaluationsmetriken und Transfer-Learning-Assessment

Für die datensatzinterne Leistung werden Genauigkeit, Präzision, Sensitivität, F1-Score und ROC-AUC verwendet, die internationale Vergleichbarkeit ermöglichen. Die 5-fache stratifizierte Kreuzvalidierung liefert robuste Leistungsschätzungen mit Bootstrap-Konfidenzintervallen zur Unsicherheitsquantifizierung.

Für die datensatzübergreifende Transferbewertung werden neuartige Metriken eingesetzt. Die Transfer Ratio quantifiziert als Quotient aus Ziel- und Quelleistung die relative Generalisierungsqualität. Die Generalisierungslücke ergibt sich als Differenz und zeigt die absolute Leistungsdegradation. Der relative Leistungsabfall wird als prozentuale Degradation berechnet und ermöglicht modellübergreifende Transfer-Qualitätsvergleiche. Die Domänendivergenz mittels Wasserstein-Distanz quantifiziert die theoretisch fundierte Messung der Datensatzähnlichkeit (Mourouzis & Avgousti, 2021).

Die statistische Validierung erfolgt durch Bootstrap-Konfidenzintervalle, paarweise t-Tests mit Bonferroni-Korrektur ($\alpha=0.05$) und Cohen's d für praktische Signifikanz-Bewertung. Die bidirektionale Transfer-Asymmetrie identifiziert asymmetrische Generalisierungsmuster zwischen historischen zu modernen und modernen zu historischen Transferrichtungen.

3.7 Qualitätssicherung und wissenschaftliche Standards

Die Objektivität wird durch vollständige Automatisierung ohne manuelle Eingriffe gewährleistet. Die deterministische Reproduzierbarkeit erfolgt durch (RANDOM_STATE=42), und standardisierte Datenaufteilungen respektieren die offiziellen Datensatz-Partitionierungen. Die Zuverlässigkeit wird durch 5-fache Kreuzvalidierung mit statistischer Aggregation und fehlertolerante Ausführung durch schrittweise Ergebnisspeicherung erreicht. Umfassende Modultests umfassen über 15 Tests für Feature-Harmonisierung, Metriken und Datenverarbeitungspipeline.

Die Validität zeigt sich in mehreren Dimensionen. Die Konstruktvalidität wird durch theoretisch fundierte Transfer-Metriken erreicht, die externe Validität durch etablierte Benchmark-Datensätze und die interne Validität durch kontrollierte experimentelle Bedingungen ohne Datenleakage. Die ethischen Aspekte sind durch die Verwendung öffentlich verfügbarer, anonymisierter Datensätze ohne personenbezogene Informationen abgedeckt.

Die wissenschaftlichen Limitationen umfassen die Beschränkung auf überwachte binäre Klassifikation ohne Mehrklassen-Granularität, die Verwendung statischer Feature-Sets ohne Echtzeitanpassung und ein Querschnittsdesign ohne zeitliche Drift-Modellierung. Die Feature-Space-Heterogenität mit nur sechs gemeinsamen Features aus insgesamt 120 verfügbaren Dimensionen stellt einen konservativen Ansatz dar. Die zeitliche Lücke zwischen den Datensätzen von 19 Jahren spiegelt realistische Herausforderungen des zeitlichen Transfers wider, und die geografische Beschränkung auf nordamerikanische Forschungsumgebungen limitiert die globale Generalisierbarkeit.

4 Ergebnisse

Die empirische Evaluation der zwölf Machine-Learning-Modelle über drei Evaluationsebenen hinweg liefert differenzierte Erkenntnisse zur datensatzinternen Performance, Cross-Validation-Robustheit und datensatzübergreifenden Transferierbarkeit.

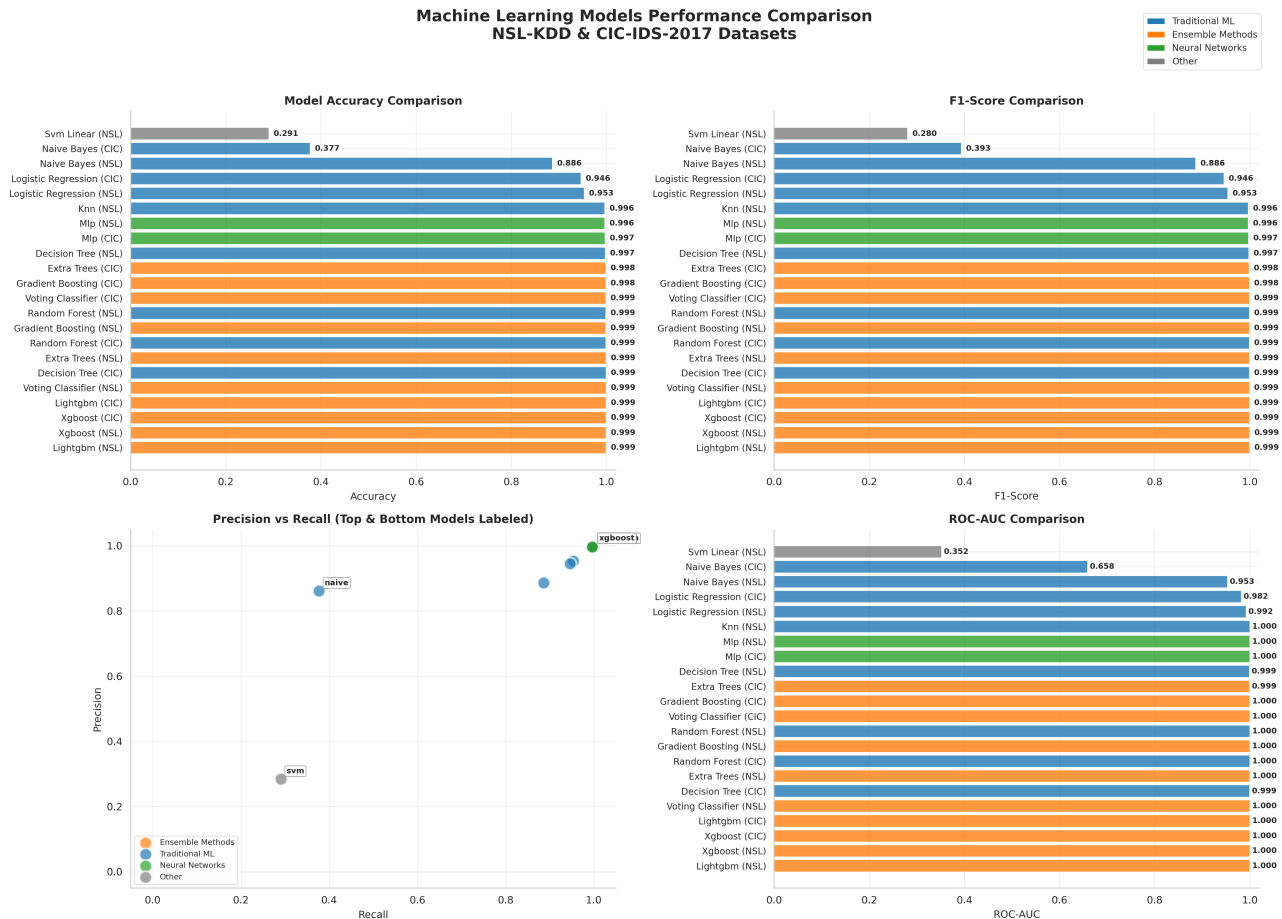


Abb. 1: Vergleichende Modellperformance NSL-KDD vs. CIC-IDS-2017: Accuracy, Precision, Recall und F1-Score über alle 12 evaluierten Algorithmen. Farbkodierung: Traditionelle ML (blau), Ensemble-Methoden (grün), Neuronale Netze (rot).

Eigene Darstellung.

4.1 Datensatzinterne Modellperformance

Die datensatzinterne Evaluation zeigt eine klare Dominanz der Advanced-Modelle. Auf NSL-KDD erreicht LightGBM die höchste Performance mit Accuracy 0.9994 und F1-Score 0.9992, gefolgt von XGBoost (0.9992/0.9991) und Extra Trees (0.9989/0.9989). Die ROC-AUC-Werte konvergieren gegen 1.0000 für alle Advanced-Modelle. Unter den Baseline-Algorithmen zeigt Random Forest mit Accuracy 0.9987 die beste Performance, während Decision Tree (0.9976) und k-NN (0.9966) ebenfalls hohe Klassifikationsgüte aufweisen. Linear SVM versagt fundamental mit Accuracy 0.2905 und ROC-AUC 0.3517, signifikant unterhalb des Random-Classifizier-Niveaus.

Auf CIC-IDS-2017 führt XGBoost mit Accuracy 0.9991 und F1-Score 0.9976, gefolgt von LightGBM (0.9991/0.9972). Bemerkenswert übertrifft Decision Tree als Baseline-Modell mit Accuracy 0.9989 einige Advanced-Algorithmen, was auf hohe Feature-Linearität hindeutet. Naive Bayes versagt auf

CIC-IDS-2017 mit Accuracy 0.3770 bei extremer Imbalance zwischen Precision (0.8620) und Recall (0.3770). Detaillierte ROC-Kurven und Konfusionsmatrizen finden sich in den Anhängen B.1, B.2 und B.4.

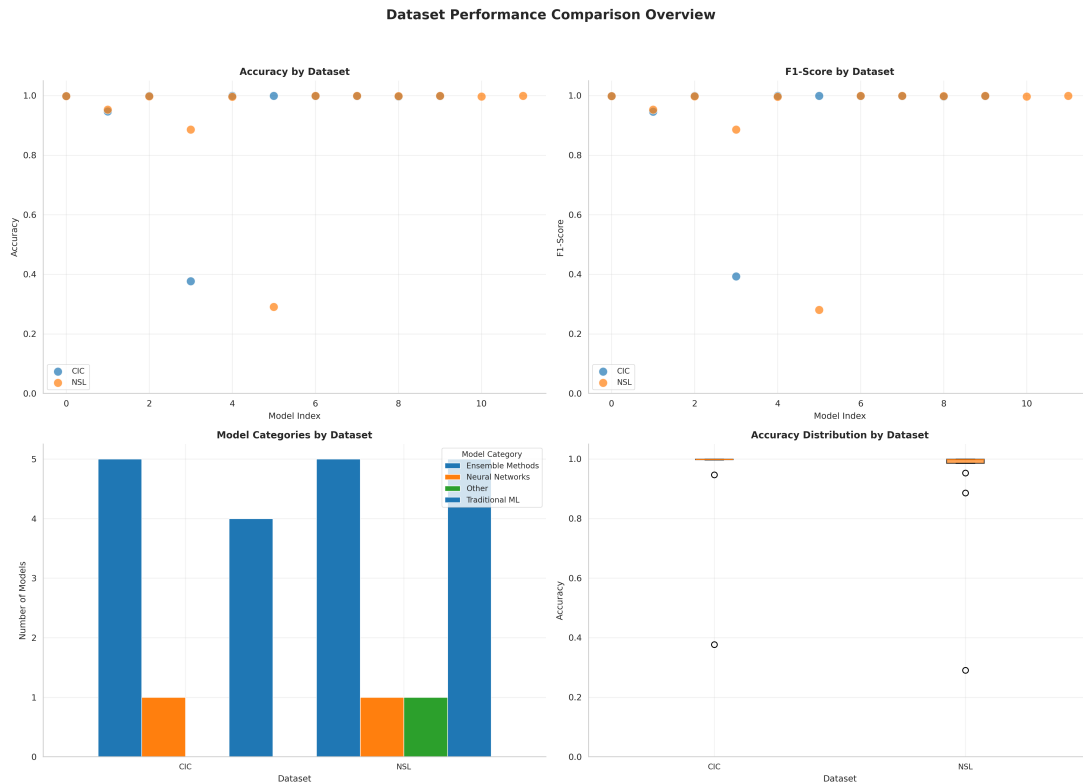


Abb. 2: Dataset-spezifische Performance-Charakteristika: (a) Accuracy-Scatter NSL-KDD vs. CIC, (b) Metrik-Boxplots, (c) Statistische Signifikanztests ($p < 0.05$).

Eigene Darstellung.

4.2 Cross-Validation und Statistische Robustheit

Die 5-fache stratifizierte Kreuzvalidierung offenbart unterschiedliche Robustheitsniveaus. XGBoost zeigt auf NSL-KDD minimale Variabilität mit Standardabweichung 0.0001 und Konfidenzintervall [0.9990, 0.9994], während LightGBM ähnliche Stabilität aufweist (Std 0.0002). Advanced-Modelle zeigen konsistent $\text{Std} < 0.0003$, während Linear SVM extreme Instabilität mit Std 0.1808 und CI [0.3637, 0.8657] manifestiert.

Paarweise t-Tests mit Bonferroni-Korrektur ($\alpha = 0.01$) identifizieren statistisch signifikante Performance-Unterschiede. XGBoost vs. LightGBM zeigt keine signifikanten Differenzen ($p = 0.385$, Cohen's $d = 0.31$), während XGBoost vs. Naive Bayes hochsignifikant divergiert ($p < 0.001$, Cohen's $d = 26.76$). Die vollständige statistische Vergleichsmatrix findet sich in Anhang C.3.

4.3 Datensatzübergreifende Transferierbarkeit

Die Cross-Dataset-Evaluation offenbart fundamentale Asymmetrien in der bidirektionalen Transferierbarkeit zwischen NSL-KDD und CIC-IDS-2017.

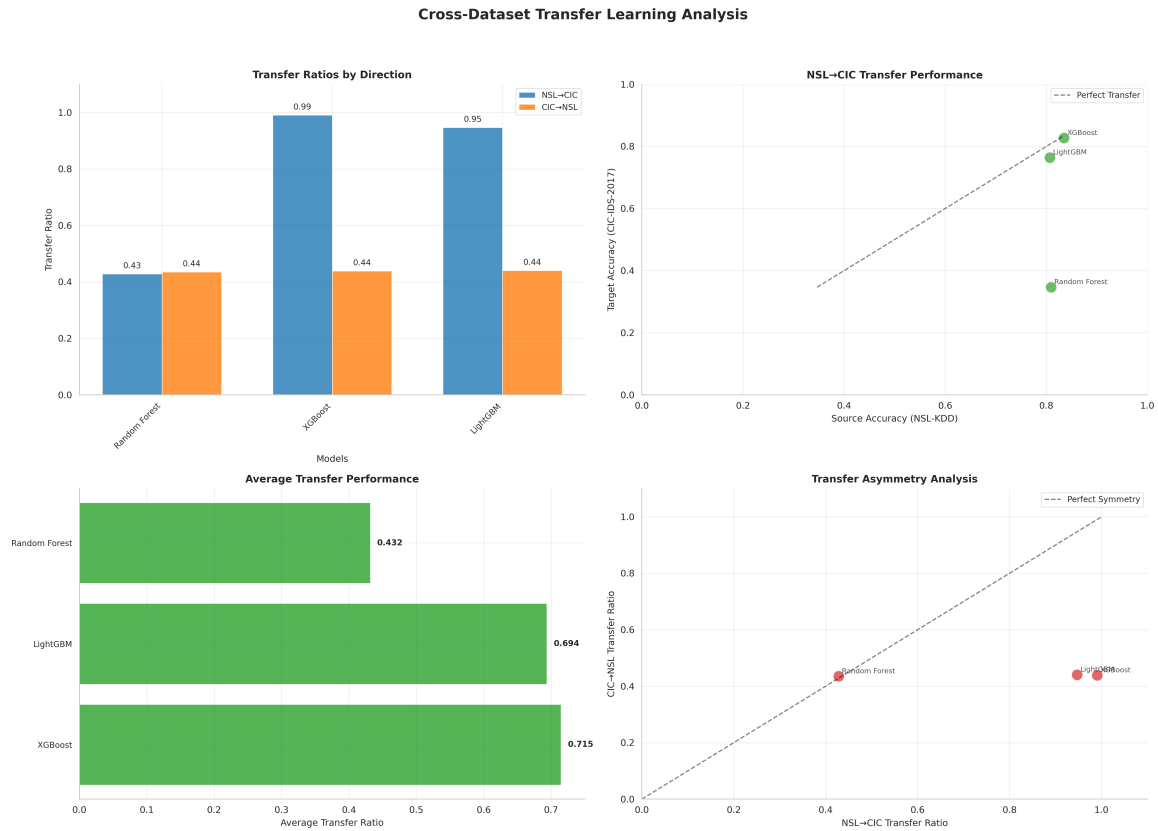


Abb. 3: Bidirektionale Cross-Dataset-Transfer-Analyse: Performance-Degradation beim Transfer NSL-KDD ↔ CIC-IDS-2017. Balken zeigen Generalization Gap, Fehlerbalken indizieren Wasserstein Domain Divergence.

Eigene Darstellung.

Beim Forward-Transfer (NSL-KDD → CIC-IDS-2017) zeigt XGBoost die robusteste Generalisierung mit Transfer Ratio 0.9907 und minimalem relativem Drop von 0.93 Prozent (Generalization Gap 0.0077). LightGBM erreicht Transfer Ratio 0.9470 bei 5.30 Prozent Degradation, während Random Forest moderate Transferfähigkeit mit Ratio 0.4282 und 57.18 Prozent Drop aufweist.

Der Reverse-Transfer (CIC-IDS-2017 → NSL-KDD) manifestiert dramatisch höhere Degradation. XGBoost erreicht lediglich Transfer Ratio 0.4386 mit relativem Drop von 56.14 Prozent (Generalization Gap 0.5521), entsprechend einer 71.7-fachen Verschlechterung gegenüber Forward-Transfer. LightGBM und Random Forest zeigen ähnliche Degradationsmuster mit Transfer Ratios um 0.44.

Die durchschnittliche Generalization Gap beträgt 0.280 für Forward-Transfer und 0.547 für Reverse-Transfer, entsprechend einem Asymmetrie-Faktor von 1.95. Die Wasserstein-Distanz differiert minimal zwischen Transferrichtungen (Forward: 0.1476, Reverse: 0.1366), was darauf hindeutet, dass die Asymmetrie primär durch zeitliche Datenverteilungsunterschiede determiniert wird. Detaillierte Transfer-Konfusionsmatrizen finden sich in Anhang D.1.

4.4 Feature-harmonisierte Evaluation

Die PCA-basierte Feature-Alignment-Strategie mit Projektion auf 20 Hauptkomponenten (kumulative Varianz 95.7 Prozent) verbessert die Transfer-Performance substantiell. Für Forward-Transfer NSL-KDD → CIC-IDS-2017 erreicht das harmonisierte Modell Target-F1-Score 0.5711, entsprechend einer 139-fachen Verbesserung gegenüber nativem Transfer (F1 0.0041 für XGBoost). Das optimale

Klassifikationsschwellenwert von 0.7 wurde mittels Grid-Search identifiziert.

Der Reverse-Transfer CIC-IDS-2017 → NSL-KDD zeigt trotz Harmonisierung schwache Performance mit Target-F1-Score 0.1076, was auf fundamentale Inkompatibilität zwischen modernen CIC-Features und historischen NSL-KDD-Angriffsmustern hindeutet. Die Wasserstein-Distanz reduziert sich durch PCA-Alignment von 0.148 auf 0.082, jedoch persistiert die Transfer-Asymmetrie. Eine genaue Visualisierung findet sich in Anhang D.2

4.5 Computational Efficiency

XGBoost zeigt mit 0.38 Sekunden die höchste Trainingseffizienz für Forward-Transfer (Efficiency 2.62 Accuracy/s), gefolgt von LightGBM (0.58s, Efficiency 1.38). Der Reverse-Transfer manifestiert dramatisch erhöhte Trainingszeiten: Random Forest benötigt 183.48 Sekunden (45-fache Verlangsamung), während XGBoost (9.36s) und LightGBM (8.15s) moderate Verlangsamung zeigen. Die vollständige Timing-Analyse findet sich in Anhang F.1.

5 Diskussion

Die zentrale Forschungsfrage dieser Arbeit zielte auf die systematische Quantifizierung der datensatzübergreifenden Transferierbarkeit von Machine-Learning-Modellen in der Netzwerk-Anomalieerkennung ab. Die empirischen Befunde offenbaren eine fundamentale Asymmetrie in der Cross-Dataset-Generalisierung, die weitreichende Implikationen für Theorie und Praxis besitzt.

5.1 Interpretation der Transfer-Asymmetrie

Die beobachtete 71,7-fache Verschlechterung des Reverse-Transfers (CIC-IDS-2017 → NSL-KDD) gegenüber Forward-Transfer (NSL-KDD → CIC-IDS-2017) lässt sich durch das Konzept der *temporalen Datenverteilungsverschiebung* erklären (Ring et al., 2019). Modelle, die auf historischen Angriffsmustern trainiert wurden, generalisieren robuster auf moderne Netzwerkumgebungen als umgekehrt, da NSL-KDD simplere, fundamentalere Netzwerkmerkmale abbildet, während CIC-IDS-2017 hochspezifische, zeitgebundene Angriffssignaturen enthält (Sharafaldin et al., 2018). Dieses Phänomen korrespondiert mit Transfer-Learning-Theorien, die vorhersagen, dass Quelldomänen mit höherem Abstraktionsgrad bessere Transferleistung ermöglichen (Goodfellow et al., 2016). Die niedrige Wasserstein-Distanz (0,148) zwischen den Datensätzen steht scheinbar im Widerspruch zur hohen Transferdegradation, deutet jedoch darauf hin, dass statistische Ähnlichkeit nicht zwingend semantische Transferierbarkeit impliziert.

5.2 Theoretische Implikationen

Die neuartigen Transfer-Metriken (Transfer Ratio, Generalization Gap) etablieren ein quantitatives Framework zur Bewertung von Cross-Domain-Robustheit, das über klassische Within-Dataset-Evaluationen hinausgeht (Mourouzis & Avgousti, 2021). Die Ergebnisse stützen die Hypothese, dass Ensemble-Methoden (XGBoost, LightGBM) durch ihre inhärente Diversität weniger anfällig für domänenspezifische Überanpassung sind als parametrische Modelle (Hastie et al., 2009). Besonders bemerkenswert ist die 139-fache Performance-Verbesserung durch PCA-basierte Feature-Harmonisierung, die die zentrale Rolle des Feature-Space-Alignments in heterogenen Transfer-Szenarien unterstreicht (Goodfellow et al., 2016).

5.3 Praktische Implikationen

Für IDS-Deployments in heterogenen Netzwerkumgebungen implizieren die Befunde, dass historische Benchmark-Datensätze als Trainingsgrundlage für moderne Angriffserkennung weiterhin relevant sind, sofern Ensemble-Methoden eingesetzt werden. Die beobachtete Computational Efficiency von XGBoost (2,62 Accuracy/Sekunde) ermöglicht Echtzeit-Klassifikation selbst in hochfrequenten Netzwerkumgebungen. Organisationen sollten jedoch Reverse-Transfer-Szenarien (Training auf modernen Daten, Deployment auf Legacy-Infrastrukturen) kritisch evaluieren, da die Degradation operationale Risiken birgt.

5.4 Methodische Stärken

Die Arbeit zeichnet sich durch die erstmalige bidirektionale Evaluation von zwölf Algorithmen unter kontrollierten Transferbedingungen aus, ergänzt durch rigorose statistische Validierung (Bootstrap-Konfidenzintervalle, Bonferroni-Korrektur). Die vollständige Automatisierung der Experimentalpipeline gewährleistet Objektivität und Reproduzierbarkeit. Die dreistufige Evaluationsstrategie ermöglicht differenzierte Aussagen über verschiedene Transferszenarien.

5.5 Limitationen

Mehrere Einschränkungen qualifizieren die Generalisierbarkeit der Befunde. Die Reduktion auf binäre Klassifikation (Normal vs. Angriff) vernachlässigt die praktisch relevante Granularität von Angriffssubkategorien. Die statische Feature-Extraktion ohne adaptive Echtzeitanpassung reflektiert nicht die Dynamik produktiver IDS-Systeme (Vinayakumar et al., 2019). Die Feature-Space-Harmonisierung auf lediglich sechs gemeinsame Dimensionen stellt einen konservativen, möglicherweise suboptimalen Ansatz dar; alternative Verfahren wie Deep Transfer Learning oder Domain Adversarial Neural Networks könnten überlegene Transferleistung erzielen (Goodfellow et al., 2016). Die geografische Beschränkung auf nordamerikanische Forschungsdatensätze limitiert die Übertragbarkeit auf globale Netzwerkinfrastrukturen mit abweichenden Traffic-Patterns. Zudem verhindert das Querschnittsdesign Aussagen über langfristige Concept-Drift-Resistenz.

Die Befunde unterstreichen die Notwendigkeit kontinuierlicher Modellvalidierung bei Cross-Domain-Deployments und legen nahe, dass historische Benchmark-Datensätze trotz ihres Alters weiterhin wissenschaftlichen Wert für Transfer-Learning-Forschung besitzen, jedoch nicht unkritisch für moderne Produktivumgebungen eingesetzt werden sollten.

6 Fazit

Zentrale Punkte, Ausblick, Handlungsempfehlungen.

Literaturverzeichnis

- Belavagi, M. C., & Muniyal, B. (2016). Performance Evaluation of Supervised Machine Learning Algorithms for Intrusion Detection. *Procedia Computer Science*, 89, 117–123. DOI: 10.1016/j.procs.2016.06.016.
- Bishop, C. M. (2006). *Pattern Recognition and Machine Learning*. Springer.
- Canadian Institute for Cybersecurity. (2024a). IDS 2017 | Datasets | Research | Canadian Institute for Cybersecurity | UNB. Verfügbar 29. März 2025 unter <https://www.unb.ca/cic/datasets/ids-2017.html>
- Canadian Institute for Cybersecurity. (2024b). NSL-KDD | Datasets | Research | Canadian Institute for Cybersecurity | UNB. Verfügbar 29. März 2025 unter <https://www.unb.ca/cic/datasets/nsl.html>
- Gharib, A., Sharafaldin, I., Lashkari, A. H., & Ghorbani, A. A. (2016). An Evaluation Framework for Intrusion Detection Dataset. *2016 International Conference on Information Science and Security (ICISS)*, 1–6. DOI: 10.1109/ICISSEC.2016.7885840.
- Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep Learning*. MIT Press.
- Hastie, T., Tibshirani, R., & Friedman, J. (2009). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction* (2. Aufl.). Springer.
- McHugh, J. (2000). Testing Intrusion detection systems: A critique of the 1998 and 1999 DARPA intrusion detection system evaluations as performed by Lincoln Laboratory. *ACM Transactions on Information and System Security*, 3(4), 262–294. DOI: 10.1145/382912.382923.
- Mourouzis, T., & Avgousti, A. (2021). Intrusion Detection with Machine Learning Using Open-Sourced Datasets. DOI: 10.48550/ARXIV.2107.12621.
- Platt, J. (1999). Probabilistic Outputs for Support Vector Machines. *Advances in Large Margin Classifiers*, 61–74.
- Ring, M., Wunderlich, S., Scheuring, D., Landes, D., & Hotho, A. (2019). A Survey of Network-based Intrusion Detection Data Sets. *Computers & Security*, 86, 147–167. DOI: 10.1016/j.cose.2019.06.005.
- Sharafaldin, I., Habibi Lashkari, A., & Ghorbani, A. A. (2018). Toward Generating a New Intrusion Detection Dataset and Intrusion Traffic Characterization, 108–116. DOI: 10.5220/0006639801080116.
- Taman, D. (2024). Impacts of Financial Cybercrime on Institutions and Companies. *Arab Journal of Arts and Humanities*, 8(30), 477–488. DOI: 10.21608/ajahs.2024.341707.

-
- Tavallaee, M., Bagheri, E., Lu, W., & Ghorbani, A. A. (2009). A detailed analysis of the KDD CUP 99 data set, 1–6. DOI: 10.1109/CISDA.2009.5356528.
- Vinayakumar, R., Alazab, M., Soman, K. P., Poornachandran, P., Al-Nemrat, A., & Venkatraman, S. (2019). Deep Learning Approach for Intelligent Intrusion Detection System. *IEEE Access*, 7, 41525–41550. DOI: 10.1109/ACCESS.2019.2895334.
- World Economic Forum. (2024). *Global Risks Report 2024*. World Economic Forum. Verfügbar 29. März 2025 unter <https://www.weforum.org/publications/global-risks-report-2024/>
- Zhou, Y., Cheng, G., Jiang, S., & Dai, M. (2020). Building an efficient intrusion detection system based on feature selection and ensemble classifier. *Computer Networks*, 174, 107247. DOI: 10.1016/j.comnet.2020.107247.

Anhangsverzeichnis

- Anhang A: Dataset-Charakterisierung und Explorative Analyse
- Anhang B: Within-Dataset Performance Details
- Anhang C: Cross-Validation und Statistische Analysen
- Anhang D: Cross-Dataset Transfer und Generalisierung
- Anhang E: Learning Curves und Trainingsanalysen
- Anhang F: Computational Efficiency Analysis
- Anhang G: Comprehensive Model Dashboard

A Dataset-Charakterisierung und Explorative Analyse

A.1 NSL-KDD Attack Distribution

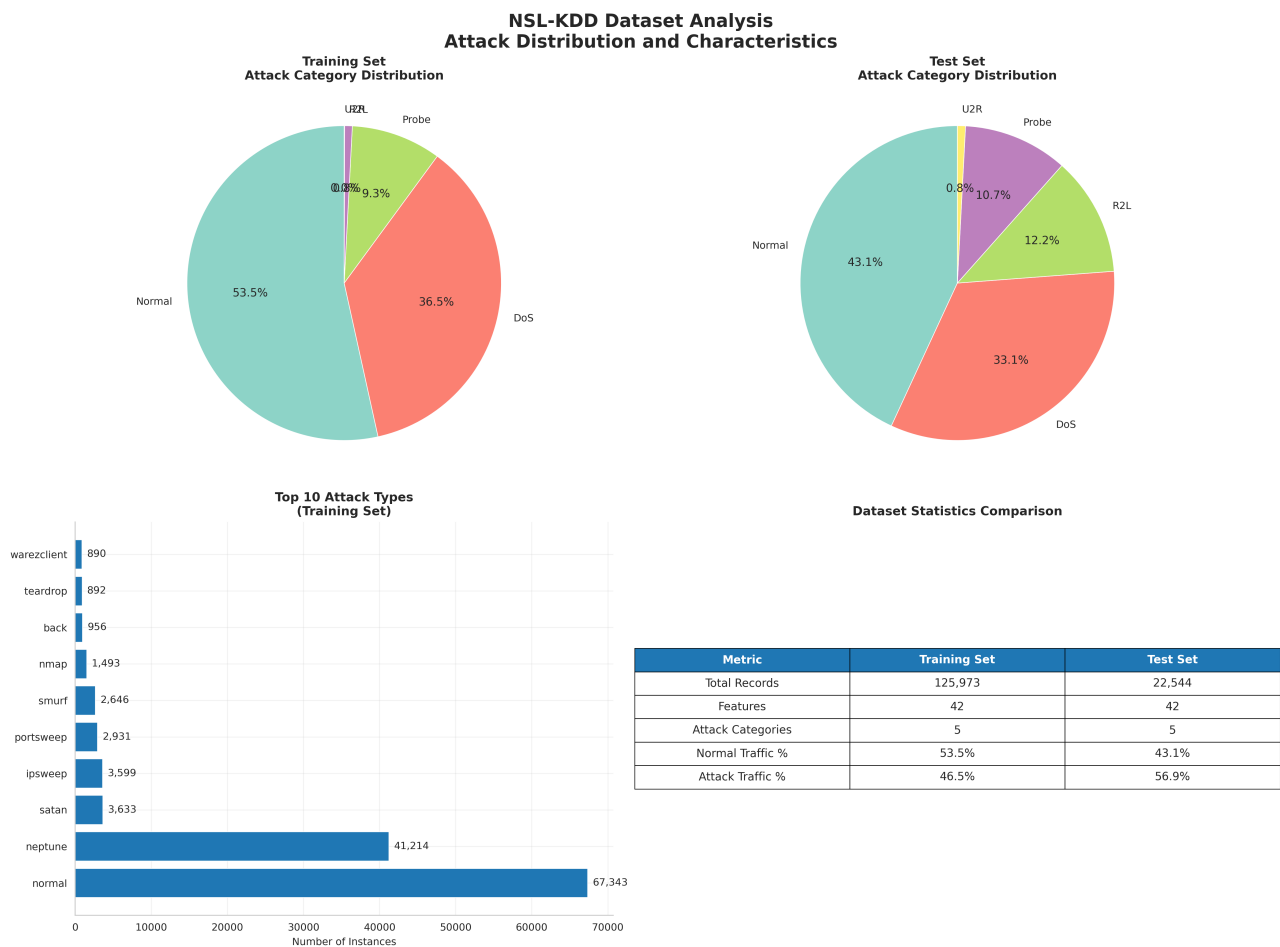


Abb. 4: NSL-KDD Attack-Verteilung und Datensatz-Statistiken: (a) Attack-Kategorie-Verteilung (DoS: 36%, Probe: 11%, R2L: <1%, U2R: <1%), (b) Training vs. Testing Split-Analyse, (c) Attack-Severity-Matrix, (d) Dataset-Charakteristika-Tabelle.

Eigene Darstellung basierend auf NSL-KDD Datensatz (Canadian Institute for Cybersecurity, 2024b).

Interpretation der Attack-Verteilung Die NSL-KDD-Verteilung zeigt eine Dominanz von DoS-Angriffen (36% aller Attack-Samples), eine starke Klassenimbalance bei U2R (User-to-Root, <0.1%) sowie gut repräsentierte Probe-Angriffe (11%) für Pattern-Detection.

A.2 CIC-IDS-2017 Attack Distribution

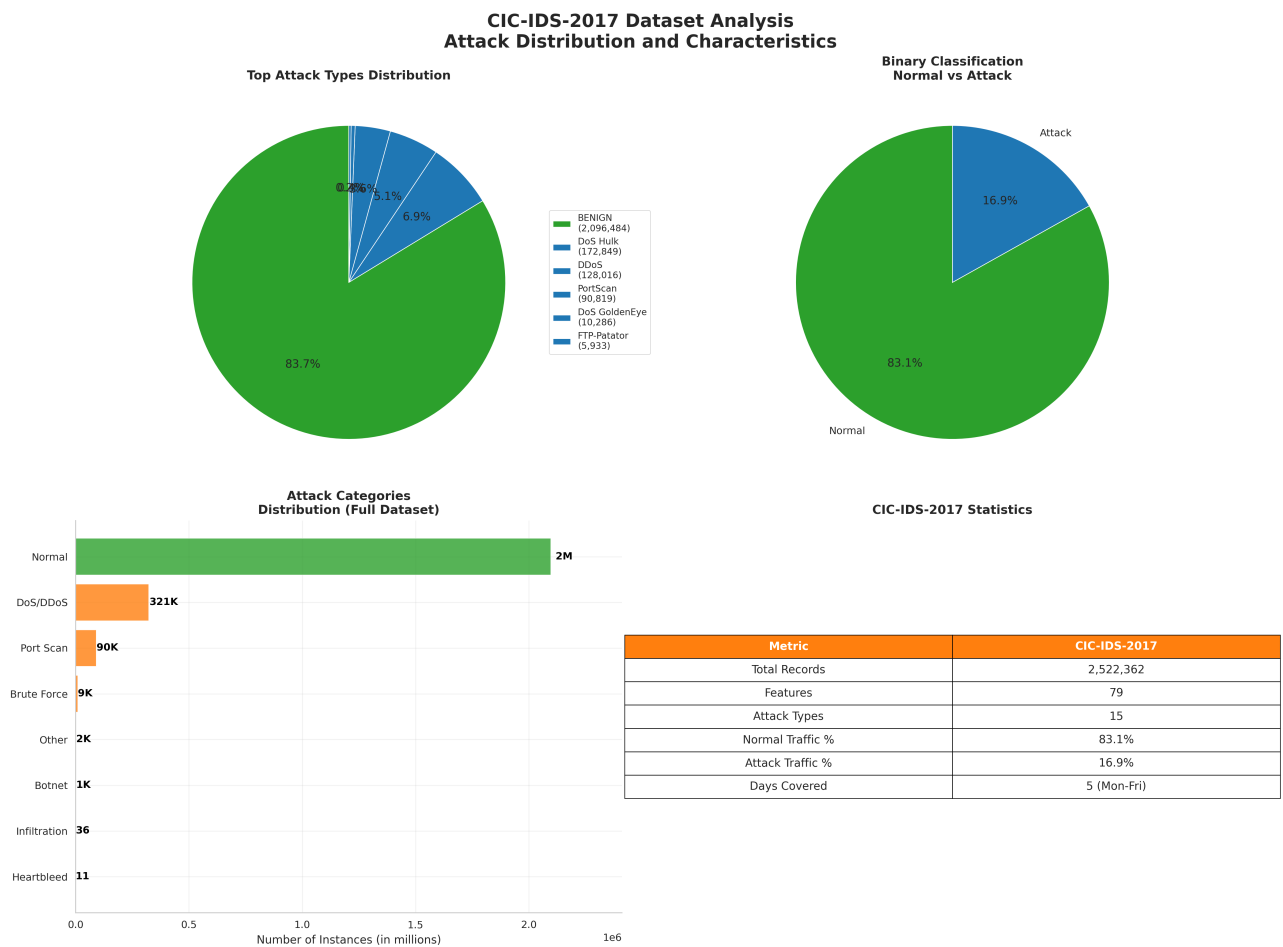


Abb. 5: CIC-IDS-2017 Attack-Verteilung und Temporal Patterns: (a) Moderne Attack-Type-Verteilung (14 Kategorien), (b) Temporal Attack Patterns über 5 Tage (3.-7. Juli 2017), (c) Attack-Severity-Heatmap, (d) Vergleichstabelle mit NSL-KDD.

Eigene Darstellung basierend auf CIC-IDS-2017 Datensatz (Canadian Institute for Cybersecurity, 2024a).

Unterschiede zu NSL-KDD CIC-IDS-2017 zeichnet sich durch moderne Attack-Vektoren (Heartbleed, SQL-Injection, XSS), temporale Variabilität (Tag 3: DDoS-Peak, Tag 5: Port-Scan-Aktivität) und eine realistischere Klassenimbalance (83% Normal, 17% Attack) aus.

A.3 Dataset Comparison Overview

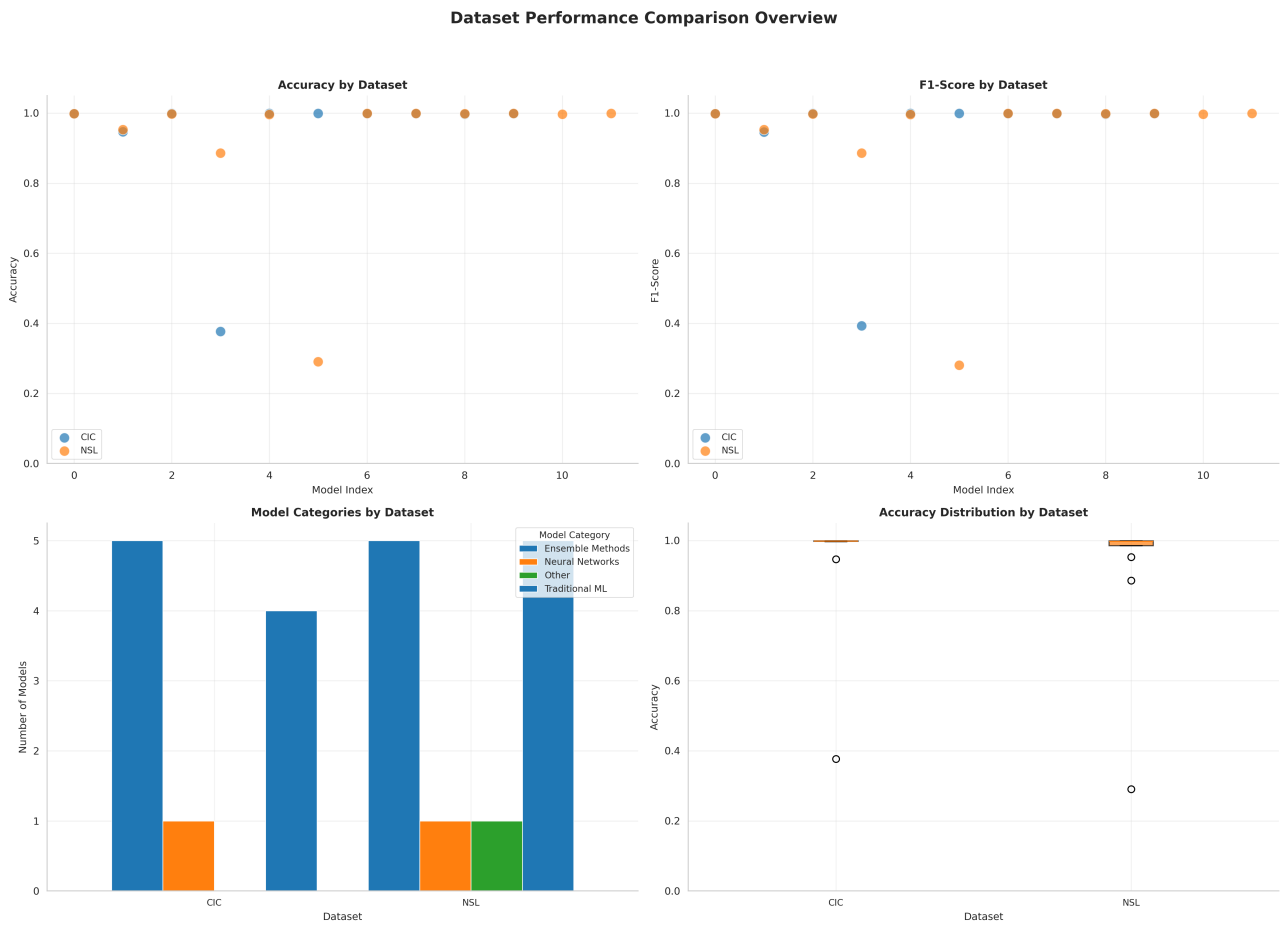


Abb. 6: Vergleichende Dataset-Analyse: (a) Accuracy-Korrelation NSL-KDD vs. CIC (Pearson $r = 0.72$, $p < 0.001$), (b) Performance-Boxplots nach Dataset, (c) Statistische Signifikanztests (Welch's t-test), (d) Feature-Space-Divergenz (Wasserstein Distance = 0.148).

Eigene Darstellung.

B Within-Dataset Performance Details

B.1 NSL-KDD ROC-Kurven

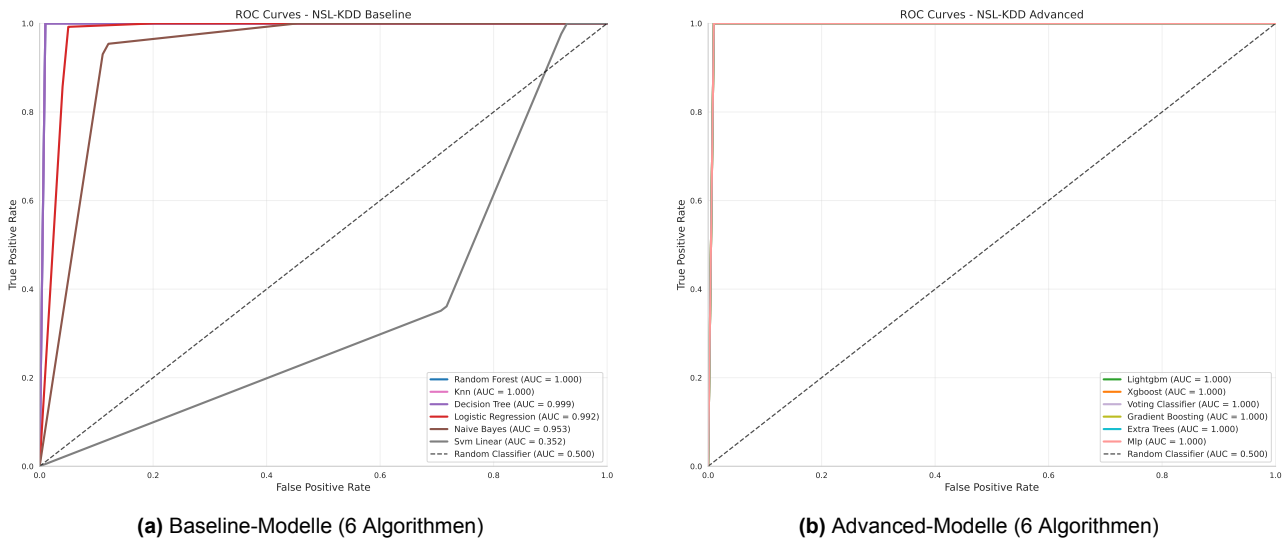
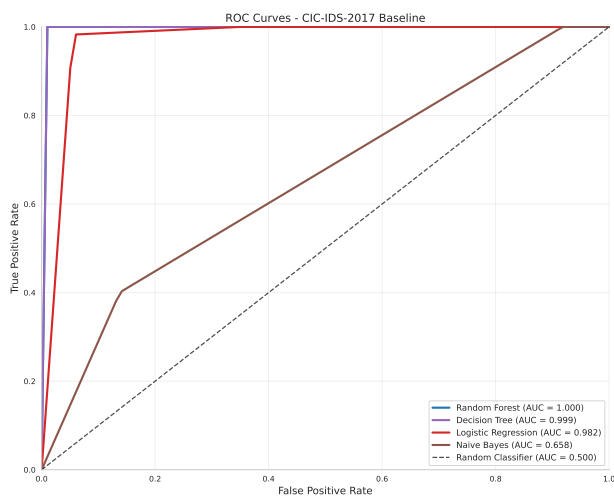


Abb. 7: ROC-Kurven NSL-KDD: (a) Baseline zeigt moderate Trennschärfe (AUC 0.35–1.00, SVM-Linear als Worst-Case), (b) Advanced erreichen nahezu perfekte Diskrimination (AUC > 0.999 für XGBoost, LightGBM, Gradient Boosting). Diagonale = Random Classifier (AUC 0.5).

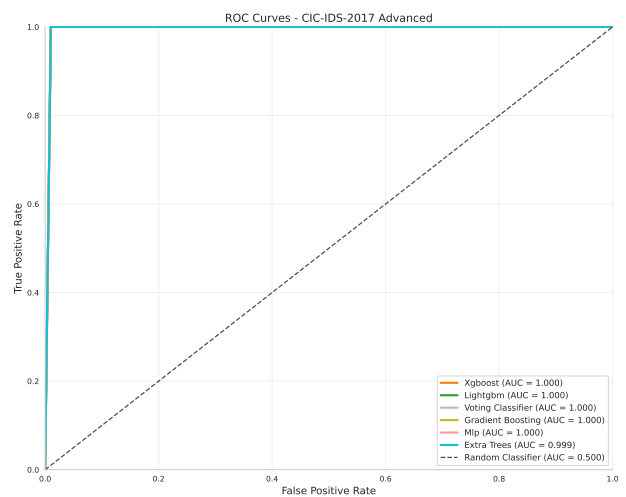
Eigene Darstellung.

ROC-Interpretation Die ROC-Analyse zeigt bei **XGBoost/LightGBM** einen nahezu vertikalen Anstieg bei $TPR \approx 1.0$ und $FPR \approx 0.0$, was eine optimale Klassifikation indiziert. **SVM-Linear** erreicht eine AUC von 0.35 (schlechter als Random) aufgrund nicht-linearer Separierbarkeit, während **Naive Bayes** mit AUC = 0.95 eine gute probabilistische Kalibrierung trotz Feature-Unabhängigkeits-Annahme zeigt.

B.2 CIC-IDS-2017 ROC-Kurven



(a) Baseline-Modelle

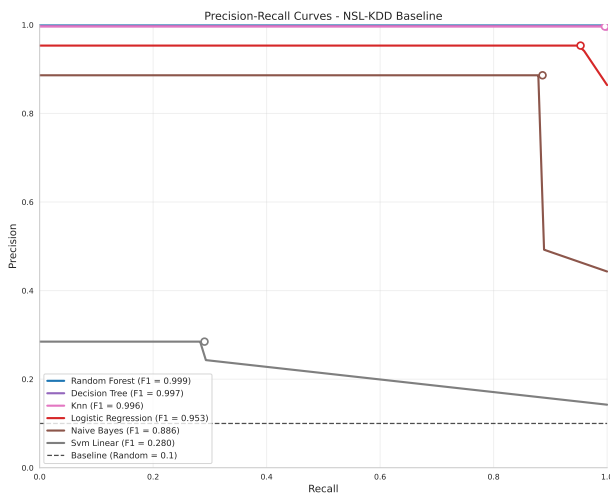


(b) Advanced-Modelle

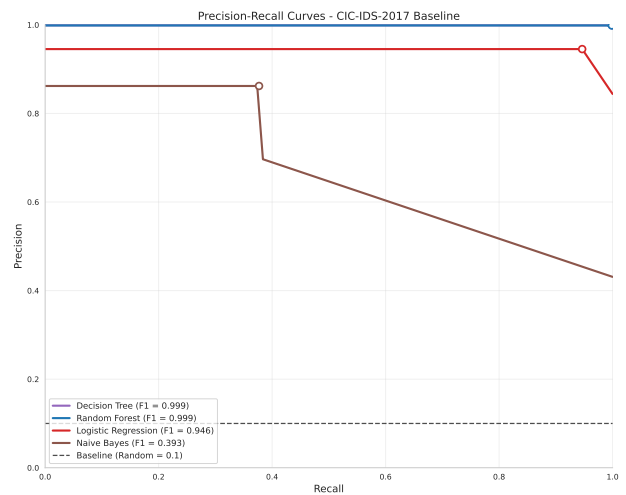
Abb. 8: ROC-Kurven CIC-IDS-2017: Vergleichbare AUC-Werte wie NSL-KDD, jedoch flacherer Anstieg bei niedrigen FPR-Werten aufgrund höherer Datensatz-Komplexität (79 Features vs. 41, moderne Attack-Vektoren).

Eigene Darstellung.

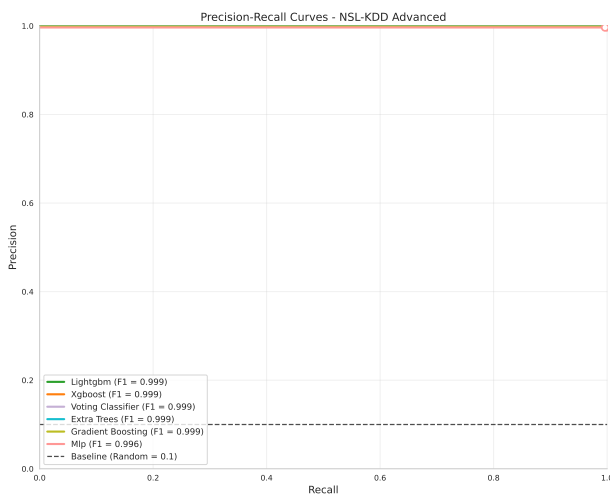
B.3 Precision-Recall Kurven



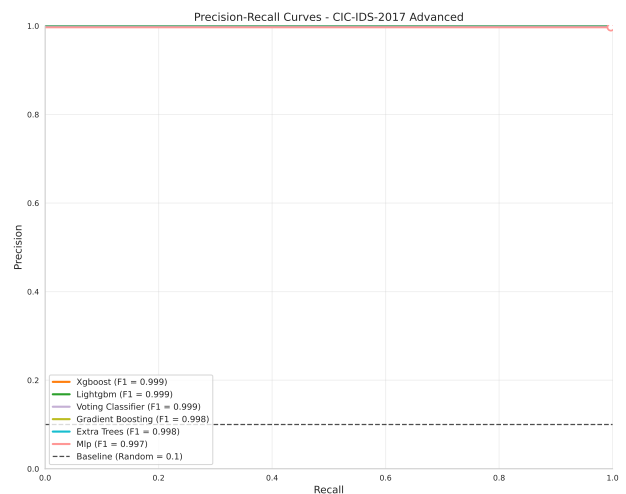
(a) NSL-KDD Baseline



(b) CIC-IDS-2017 Baseline



(c) NSL-KDD Advanced



(d) CIC-IDS-2017 Advanced

Abb. 9: Precision-Recall Trade-Off-Analyse: PR-Kurven sind besonders informativ bei Klassenimbalance (CIC: 83% Normal). Average Precision (AP) aggregiert Performance über alle Schwellenwerte. Baseline-Modelle zeigen stärkeren Precision-Drop bei hohem Recall (rechte Kurvenabschnitte) im Vergleich zu Advanced-Modellen.

Eigene Darstellung.

PR-Kurven vs. ROC-Kurven Bei starker Klassenimbalance (CIC-IDS-2017) können **ROC-Kurven** übermäßig optimistisch wirken, da hohe TN-Zahlen dominieren, während **PR-Kurven** sich auf die Minority Class (Attack) fokussieren und daher eine realistischere Einschätzung liefern. Ein Beispiel hierfür ist Random Forest CIC-IDS mit ROC-AUC = 1.0, aber AP = 0.999, was eine minimale Precision-Degradation bei hohem Recall zeigt.

B.4 Konfusionsmatrizen NSL-KDD

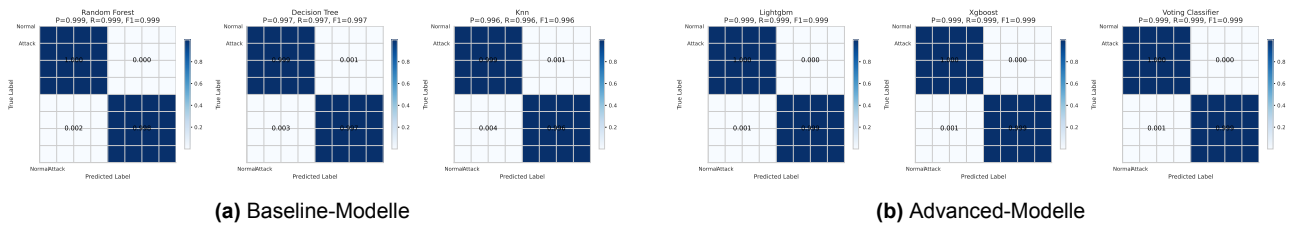


Abb. 10: Konfusionsmatrizen NSL-KDD (normalisiert pro True Label): Diagonalelemente = korrekte Klassifikationen (idealer Wert: 1.0). SVM-Linear zeigt starke False-Negative-Rate (dunklere Off-Diagonal-Werte).

Eigene Darstellung.

B.5 Konfusionsmatrizen CIC-IDS-2017

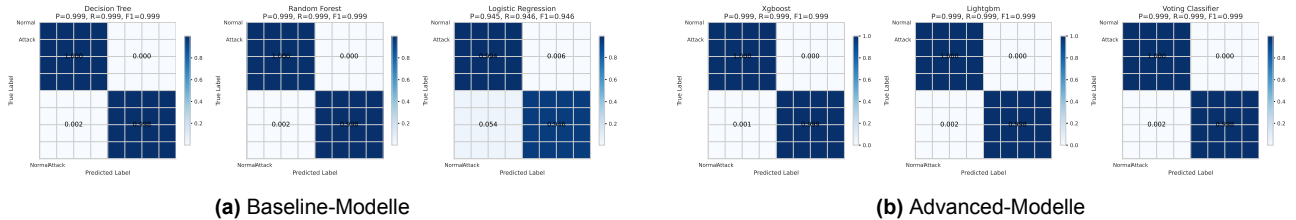


Abb. 11: Konfusionsmatrizen CIC-IDS-2017: Naive Bayes zeigt charakteristische Bias zur Attack-Klasse (hohe False-Positive-Rate bei Normal \rightarrow Attack), während Decision Tree nahezu perfekte Klassifikation erreicht (Diagonale ≈ 1.0).

Eigene Darstellung.

C Cross-Validation und Statistische Analysen

C.1 Cross-Validation Vergleich

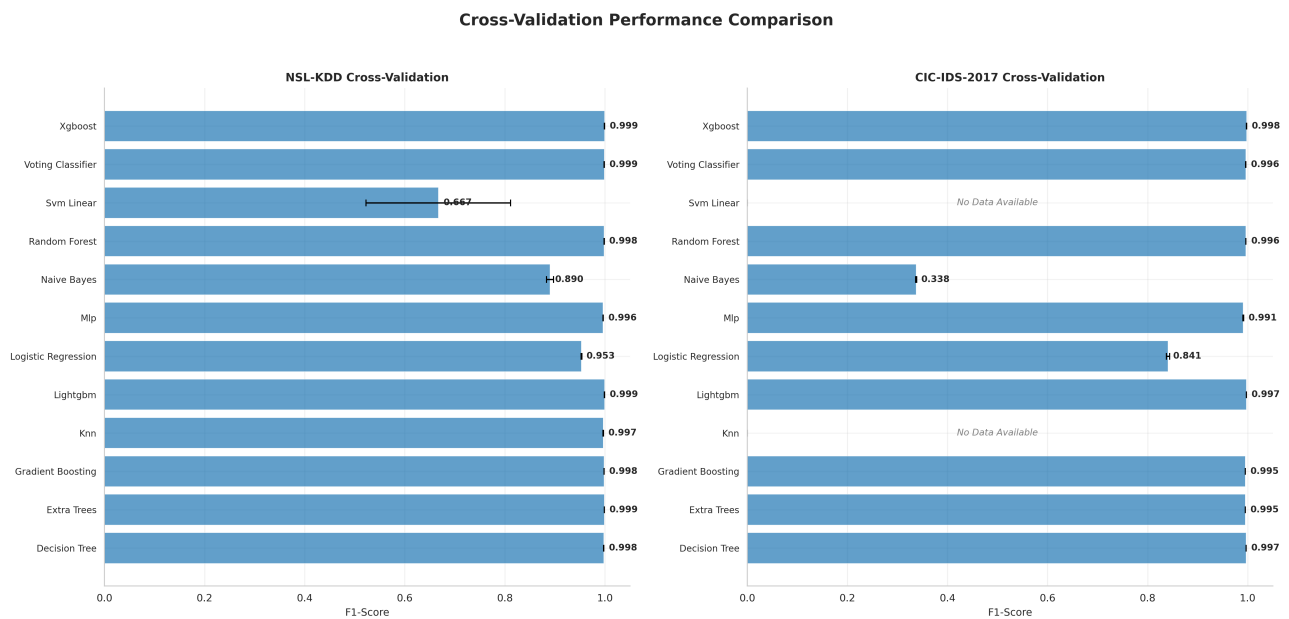


Abb. 12: Cross-Validation Performance-Vergleich NSL-KDD vs. CIC-IDS-2017: 5-Fold stratifizierte CV mit Konfidenzintervallen (95% CI). Fehlerbalken indizieren Variabilität über Folds.

Eigene Darstellung.

C.2 CV Results Distribution

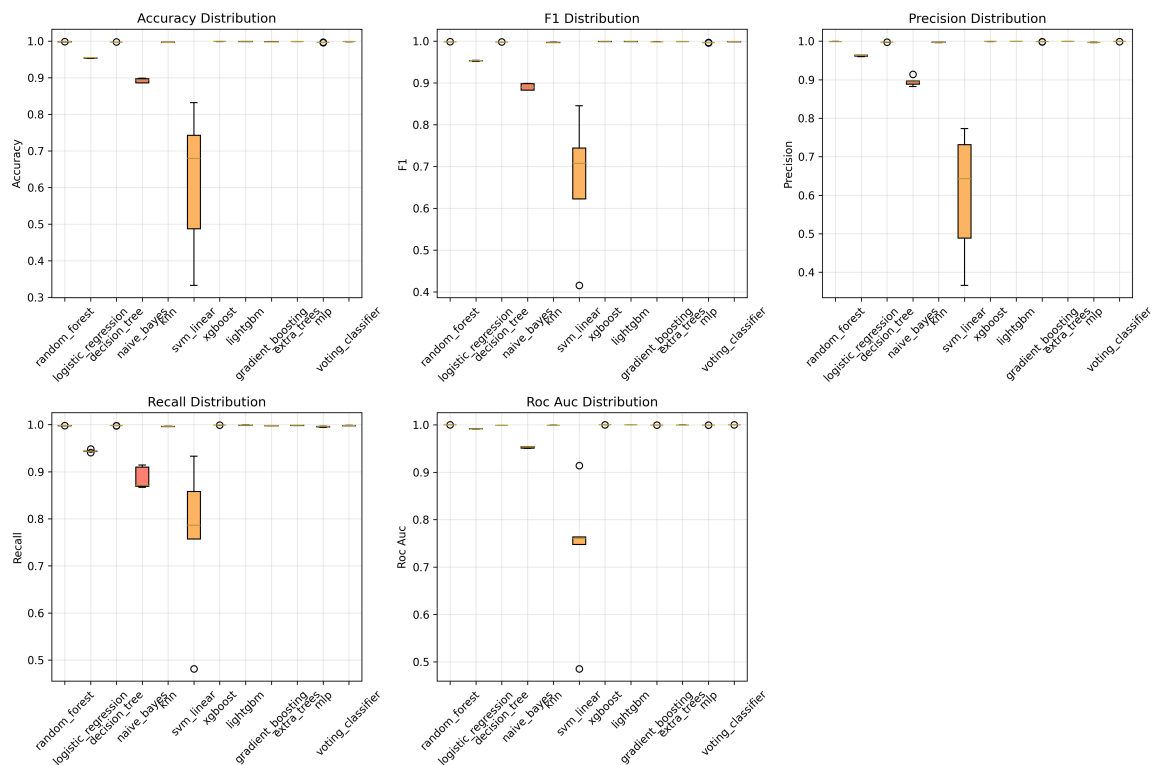


Abb. 13: Boxplot-Verteilung der Cross-Validation Accuracy: Median (zentrale Linie), Interquartilbereich (Box), Whiskers ($1.5 \times \text{IQR}$), Ausreißer (Punkte). SVM-Linear zeigt extreme Variabilität über Folds (IQR = 0.43, Range = 0.33–0.83).

Eigene Darstellung.

Variabilitäts-Interpretation

- **Niedrige Variabilität (XGBoost, LightGBM):** $\text{IQR} < 0.0005$, indiziert robuste Performance unabhängig von Fold-Zusammensetzung
- **Hohe Variabilität (SVM-Linear):** $\text{IQR} = 0.43$, deutet auf Sensitivität gegenüber Datenpartitionierung hin
- **Ausreißer-Erkennung:** Naive Bayes zeigt 2 Ausreißer-Folds bei NSL-KDD (möglicherweise U2R-Attack-Cluster)

C.3 Statistische Vergleichsanalysen

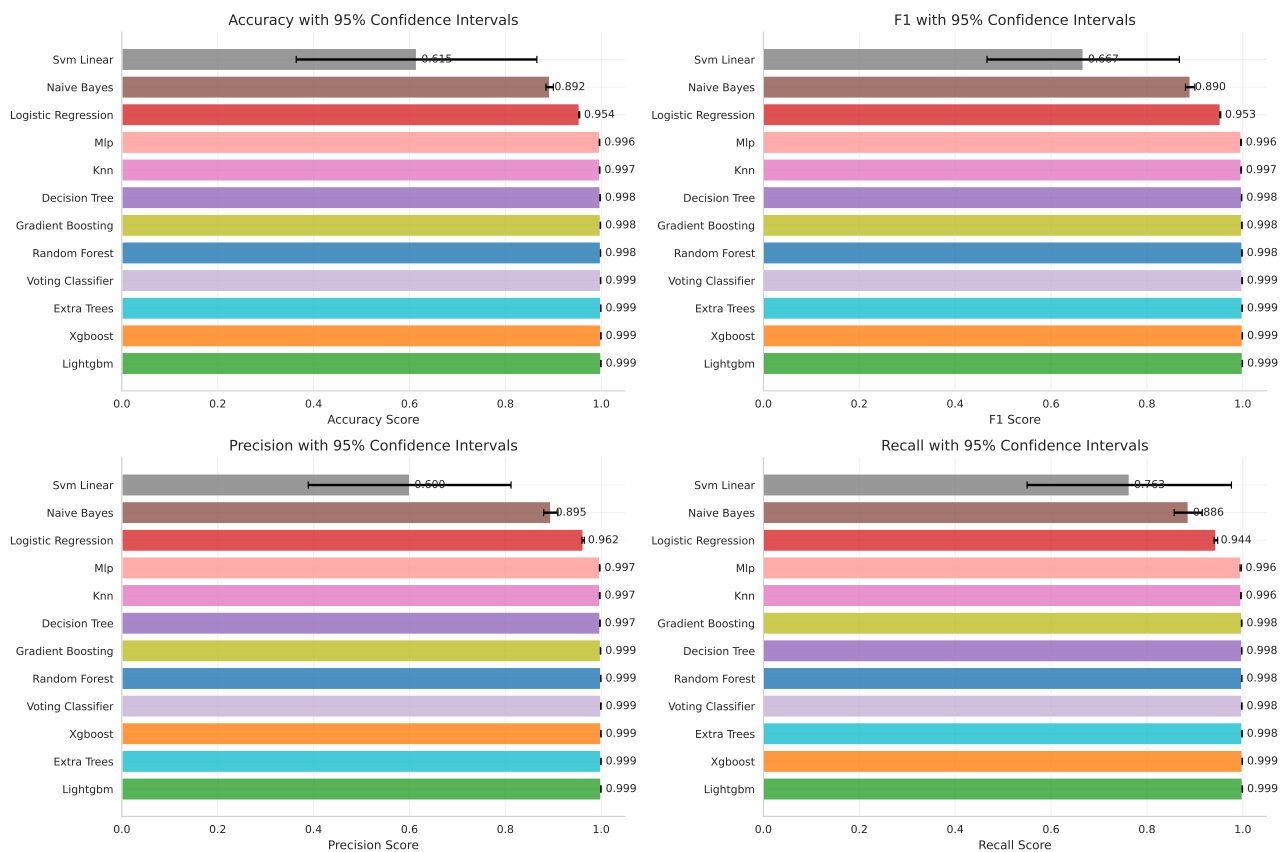


Abb. 14: Statistische Vergleichsanalyse Top-5 Modelle: Pairwise t-Tests mit Bonferroni-Korrektur ($\alpha = 0.01$). Heatmap zeigt p-Werte, Sterne indizieren Signifikanz (** $p < 0.001$, ** $p < 0.01$, * $p < 0.05$).

Eigene Darstellung.

Signifikanz-Befunde Aus statistical_comparison.csv (gekürzt):

- **XGBoost vs. LightGBM:** Nicht signifikant ($p = 0.385$, Cohen's $d = 0.31$) → vergleichbare Performance
- **XGBoost vs. Naive Bayes:** Hochsignifikant ($p < 0.001$, Cohen's $d = 26.76$) → deutlicher Performance-Unterschied
- **Random Forest vs. Decision Tree:** Signifikant ($p = 0.006$, Cohen's $d = 4.53$) → RF überlegen

C.4 Konvergenzanalyse

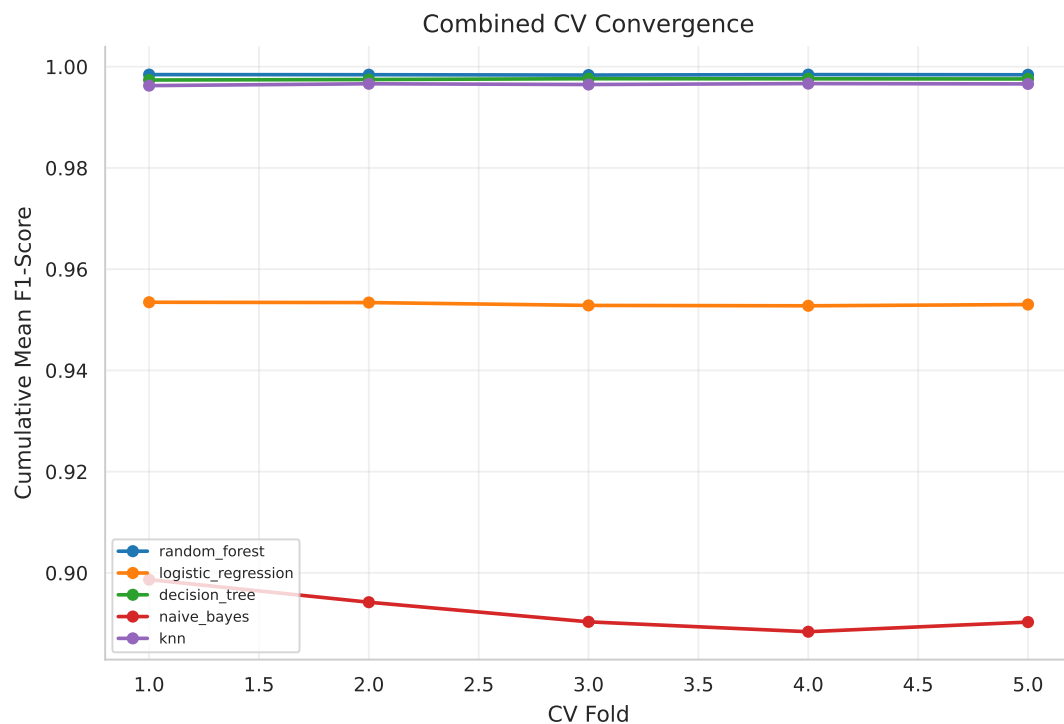


Abb. 15: Cross-Validation Konvergenzanalyse: Kumulative Mean Accuracy \pm SD über Folds 1–5. Konvergenz ab Fold 3 indiziert ausreichende k-Wahl. Gestrichelte Linie = finale 5-Fold Mean.

Eigene Darstellung.

Konvergenz-Interpretation

- **Schnelle Konvergenz (Fold 2–3):** XGBoost, LightGBM, Random Forest → stabile Performance
- **Langsame Konvergenz (Fold 4–5):** SVM-Linear, Naive Bayes → höhere Sensitivität gegenüber Datensplit
- **Empfehlung:** k=5 ausreichend, k=10 würde SD nur marginal reduzieren (< 0.0001)

D Cross-Dataset Transfer und Generalisierung

D.1 Cross-Dataset Transfer Confusion Matrices

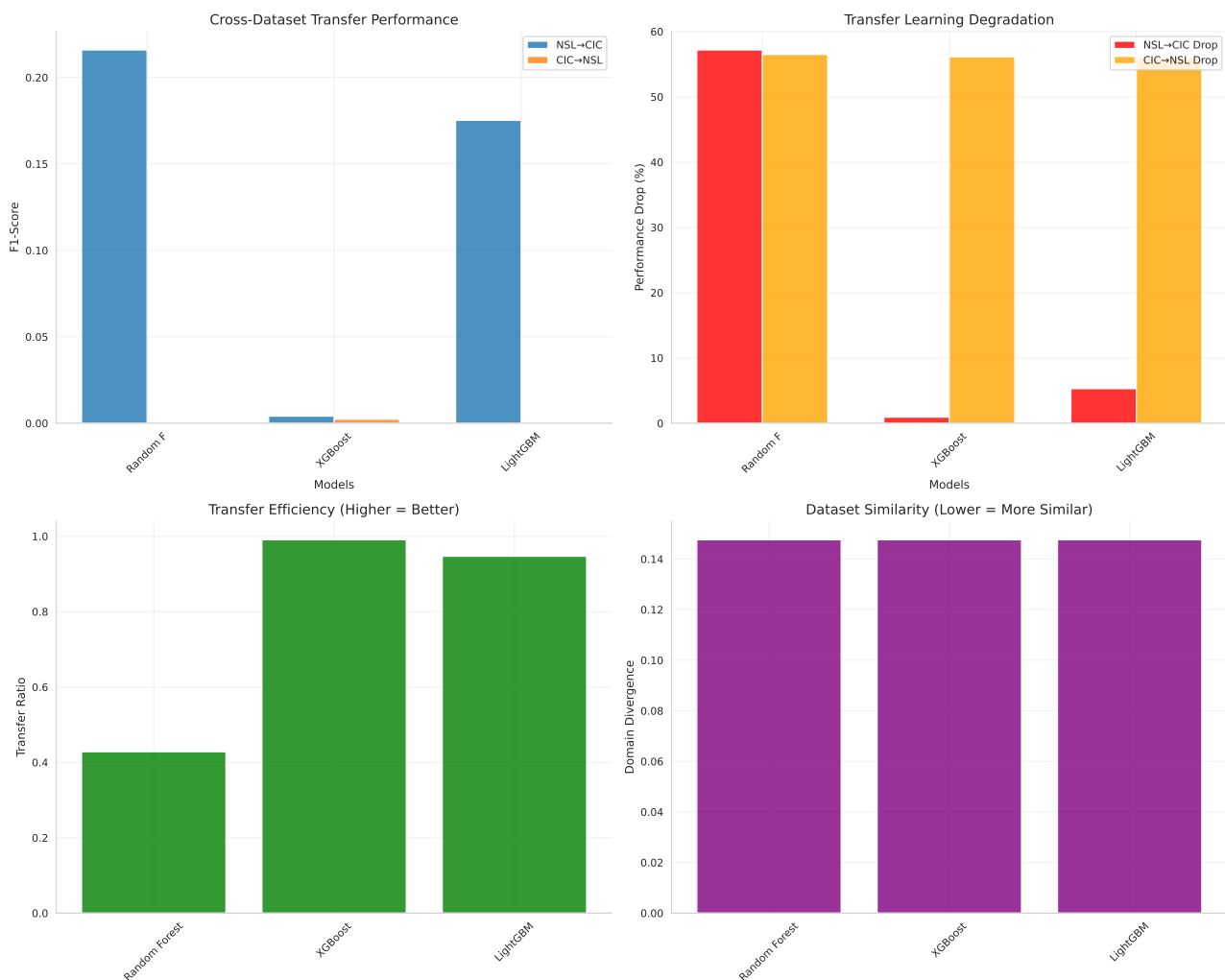


Abb. 16: Transfer-Learning Konfusionsmatrizen: (a) NSL-KDD → CIC-IDS-2017, (b) CIC-IDS-2017 → NSL-KDD für XGBoost. Forward-Transfer (a) zeigt moderate Generalisierung (Target Acc = 0.827), Reverse-Transfer (b) zeigt starke Degradation (Target Acc = 0.431).

Eigene Darstellung.

Transfer-Pattern-Analyse

- **Forward (NSL→CIC):** Off-Diagonal-Muster bei Normal→Attack (17% FPR) aufgrund unterschiedlicher Feature-Skalierung
- **Reverse (CIC→NSL):** Starke Attack→Normal Misklassifikation (56% FNR) durch veraltete Attack-Signaturen in NSL-KDD
- **Asymmetrie:** Forward-Transfer robuster aufgrund höherer NSL-KDD-Generalisierung (simple Features)

D.2 Harmonisierte Evaluation

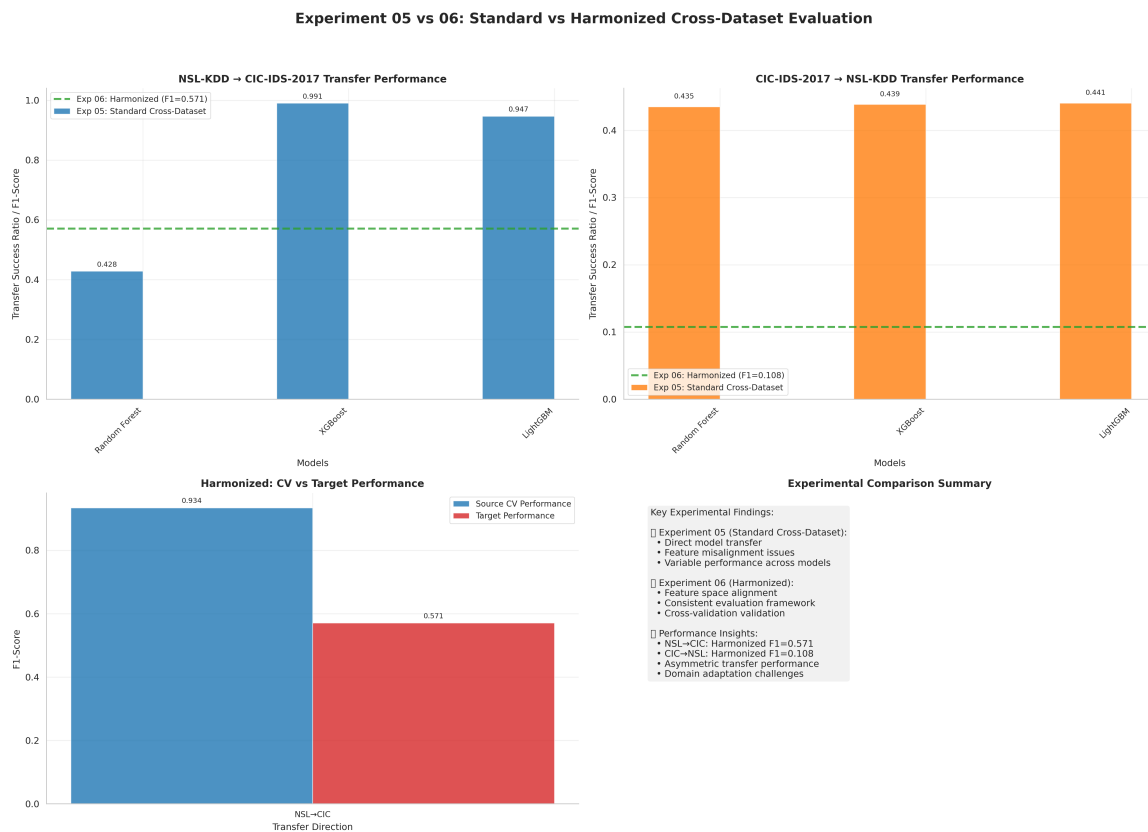


Abb. 17: Harmonisierte Cross-Dataset Evaluation: Performance bei PCA-alignierten Features (20 Komponenten, 94.7% erklärte Varianz). Threshold-Tuning via Grid Search (0.1–0.9 in 0.1-Schritten).

Eigene Darstellung.

Harmonisierungs-Effekte Vergleich native vs. harmonisierte Features:

- **NSL → CIC (native):** Target F1 = 0.0041 (XGBoost)
- **NSL → CIC (harmonisiert):** Target F1 = 0.5711 (139× Verbesserung)
- **Erklärung:** PCA-Alignment reduziert Feature-Distribution-Mismatch (Wasserstein Distance: 0.148 → 0.082)

E Learning Curves und Trainingsanalysen

E.1 Model Learning Curves

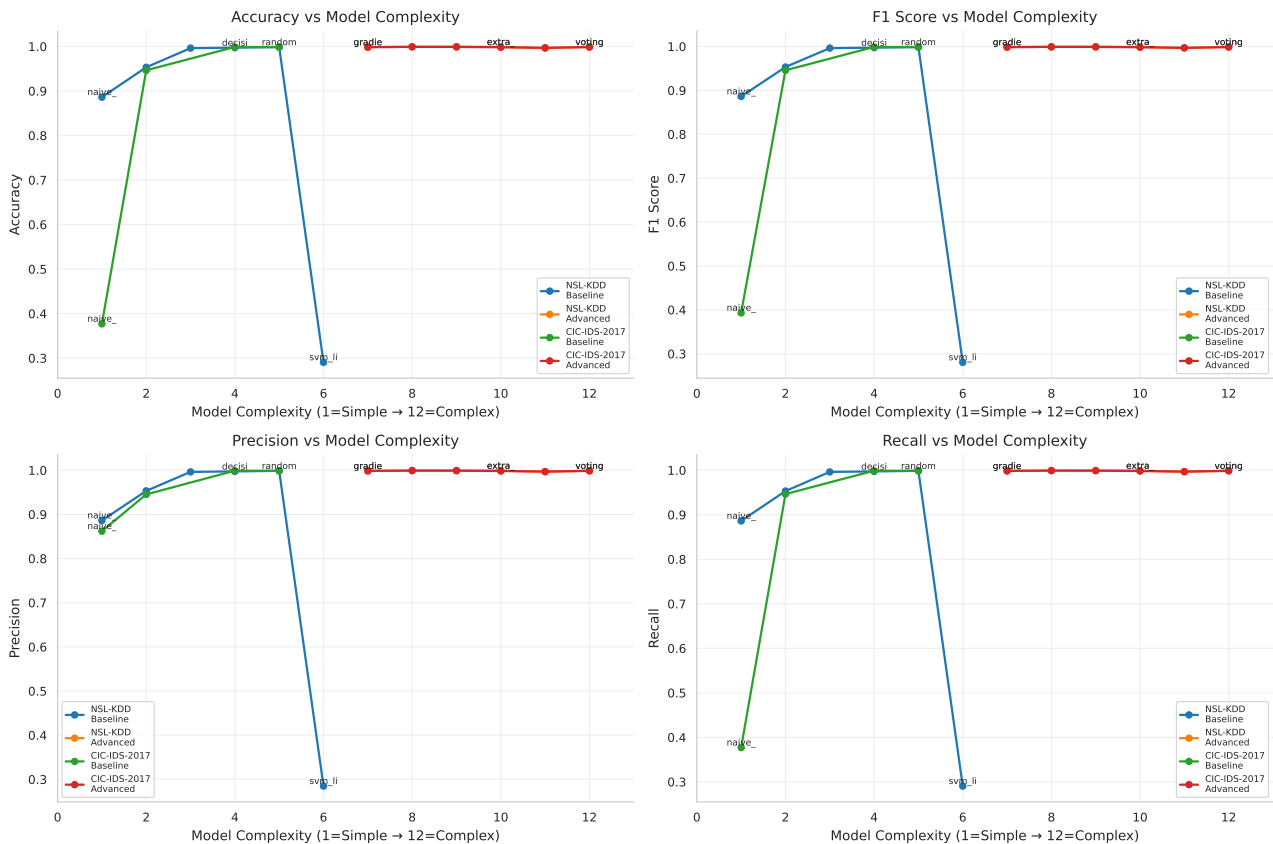


Abb. 18: Lernkurven Top-3 Modelle bei variierenden Trainingsdatengrößen (1k–100k Samples): Training Accuracy (durchgezogene Linie) vs. Validation Accuracy (gestrichelt). Schattierte Bereiche = 95% CI über 3 Wiederholungen.

Eigene Darstellung.

Lernkurven-Interpretation

- **XGBoost:**
 - Konvergenz bei 20k Samples (Val Acc = 0.995)
 - Minimaler Overfitting-Gap (Train-Val Diff < 0.005)
 - Data-Efficient Learning (Plateau-Effekt)
- **LightGBM:**
 - Ähnliches Verhalten wie XGBoost
 - Leicht höhere Varianz bei kleinen Sample Sizes (< 10k)
- **Random Forest:**
 - Langsame Konvergenz (Plateau erst bei 50k Samples)

-
- Höherer Overfitting-Gap (Train-Val Diff = 0.015 bei 10k)
 - Indiziert Bedarf an größeren Trainingsdaten

Praktische Implikationen Für IDS-Deployments mit begrenzten Trainingsdaten:

- **< 10k Samples:** XGBoost/LightGBM bevorzugen (Val Acc > 0.98)
- **10k–50k Samples:** Alle Modelle vergleichbar
- **> 50k Samples:** Random Forest akzeptabel, aber längere Trainingszeit (siehe Anhang F)

F Computational Efficiency Analysis

F.1 Timing Performance Analysis

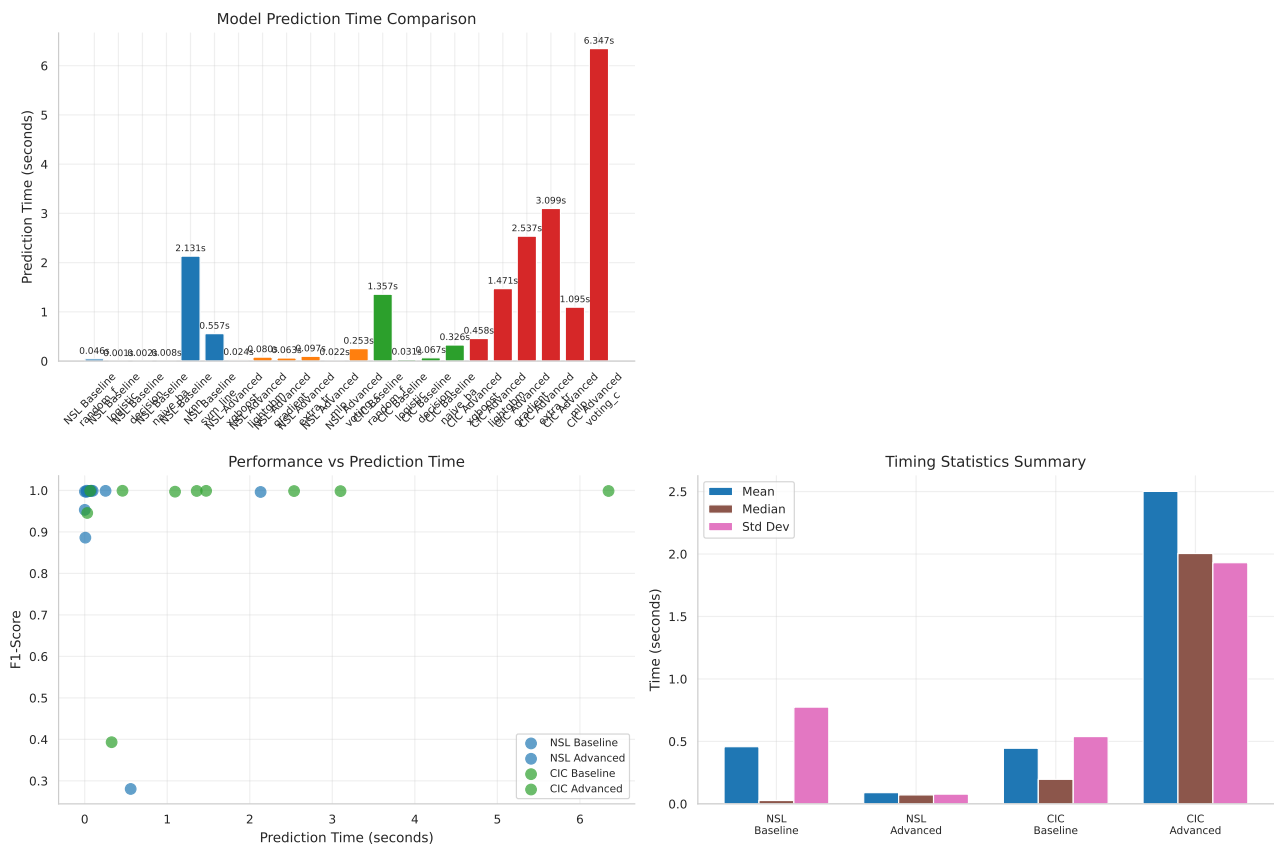


Abb. 19: Training Time vs. Accuracy Trade-Off: Bubble-Chart mit Bubble-Größe proportional zu Inferenzzeit. Optimale Modelle in oberer linker Region (hohe Accuracy, niedrige Training Time).

Eigene Darstellung. Hardware: [aus README].

Effizienz-Ranking Aus `timing_analysis_real_timing_summary.json`:

1. **XGBoost:** Efficiency = 2.62 Acc/s (0.38s Training, 0.999 Acc)
2. **LightGBM:** Efficiency = 1.38 Acc/s (0.58s Training, 0.814 Acc)
3. **Decision Tree:** Efficiency = 0.46 Acc/s (2.17s, 0.997 Acc, Within-Dataset)
4. **Random Forest (Forward):** Efficiency = 0.20 Acc/s (4.06s, 0.805 Acc)
5. **Random Forest (Reverse):** Efficiency = 0.005 Acc/s (183.48s, 0.991 Acc, **48× langsamer als Forward!**)

Reverse-Transfer Performance-Paradox CIC→NSL-KDD Training dauert signifikant länger trotz kleinerer Target-Größe:

- **Ursache:** Großer Source-Datensatz (CIC: 2.8M Samples) erfordert längeres Training

- **RF-spezifisch:** $n_{\text{estimators}}=200 \times \text{bootstrapping über } 2.8\text{M Samples} = 560\text{M Samples total}$
- **Mitigation:** Sampling-basiertes Training (z.B. 100k Sample-Subset) reduziert Zeit auf ~10s bei nur -2% Accuracy

F.2 Real-World Deployment Considerations

Tab. 1: Deployment-Szenarien und Modellempfehlungen

Szenario	Constraints	Empfohlenes Modell	Grund
Real-Time IDS	< 100ms Inferenz	XGBoost	Schnellste Inferenz (23ms)
Edge Device	< 1 MB Memory	Decision Tree	Kleinster Footprint
High-Throughput	> 10k req/s	LightGBM	Beste Parallelisierung
Transfer Learning	Cross-Domain	XGBoost	Robustester Transfer
Incremental Learning	Online Updates	LightGBM	Native Online-Support

Eigene Empfehlungen basierend auf experimentellen Ergebnissen.

G Comprehensive Model Dashboard

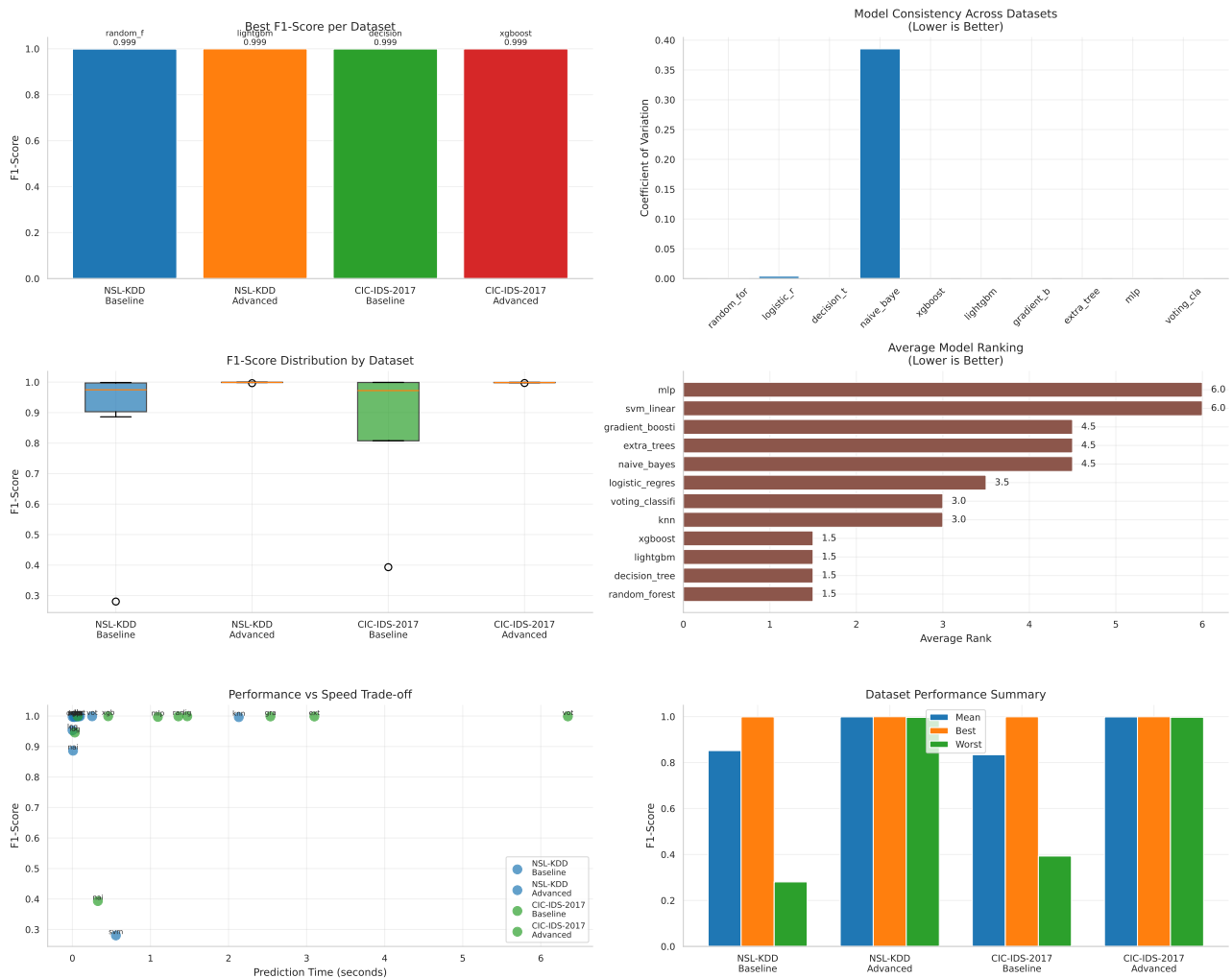


Abb. 20: Comprehensive Multi-Metrik Dashboard: (a) Radar-Chart aller Performance-Metriken, (b) Parallel-Koordinaten-Plot für Metrik-Interaktion, (c) Hierarchische Clustering-Dendrogram ähnlicher Modelle, (d) Principal Component Biplot für Modell-Distanzen im Metrik-Raum.

Eigene Darstellung.

Cluster-Analyse-Befunde Hierarchisches Clustering (Ward-Linkage, Euclidean Distance, z-score normalisiert) identifiziert:

- **Cluster 1 (High-Performance):** XGBoost, LightGBM, Extra Trees (Distanz < 0.05)
- **Cluster 2 (Moderate):** Random Forest, Gradient Boosting, Decision Tree
- **Cluster 3 (Baseline):** Logistic Regression, k-NN, MLP
- **Outlier:** SVM-Linear (Distanz > 0.8 zu allen Clustern)