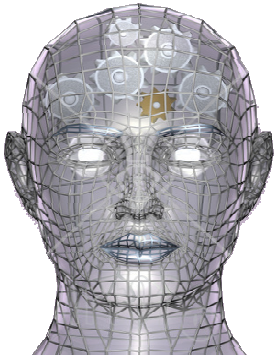


Westfälische Hochschule
Gelsenkirchen Bocholt Recklinghausen
University of Applied Sciences

Artificial Intelligence

Hype oder Trend?

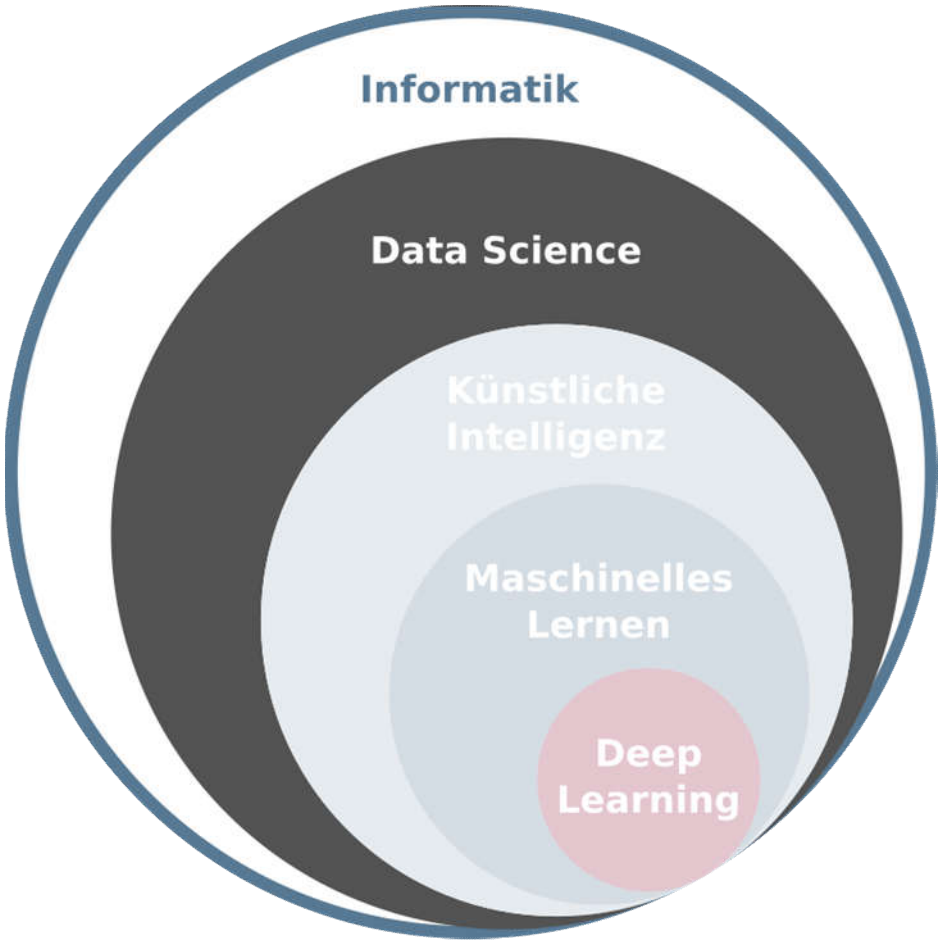
Dr.
Rolf Reinema

Prof. Dr. (TU NN)
Norbert Pohlmann

SIEMENS

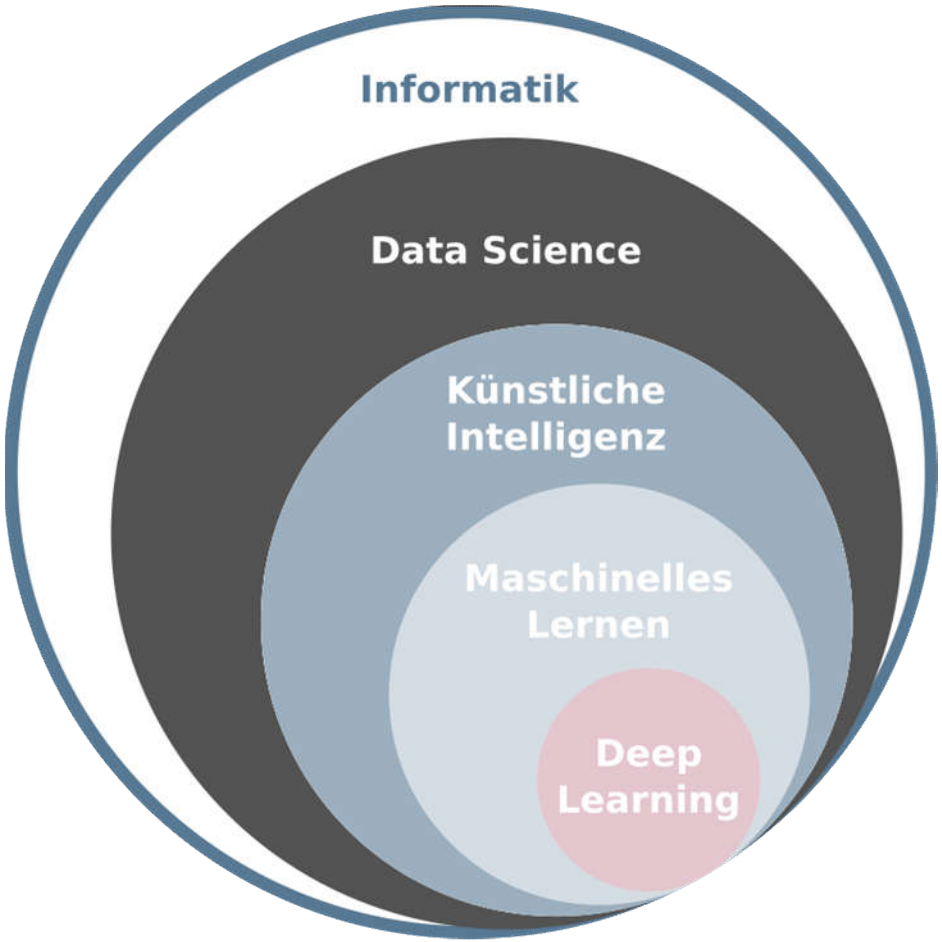
if(is)
internet-sicherheit.

Einordnung
→ Data Science

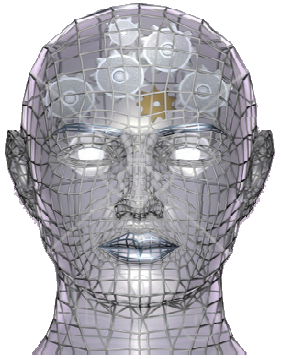


- **Data Science** bezeichnet generell die **Extraktion von Wissen aus Daten**.
- **Da es immer mehr Daten gibt, kann auch immer mehr Wissen daraus abgeleitet werden.**
- Abgrenzung zur künstlichen Intelligenz:
  - Statistiken
  - Kennzahlen
  - Datenerhebung

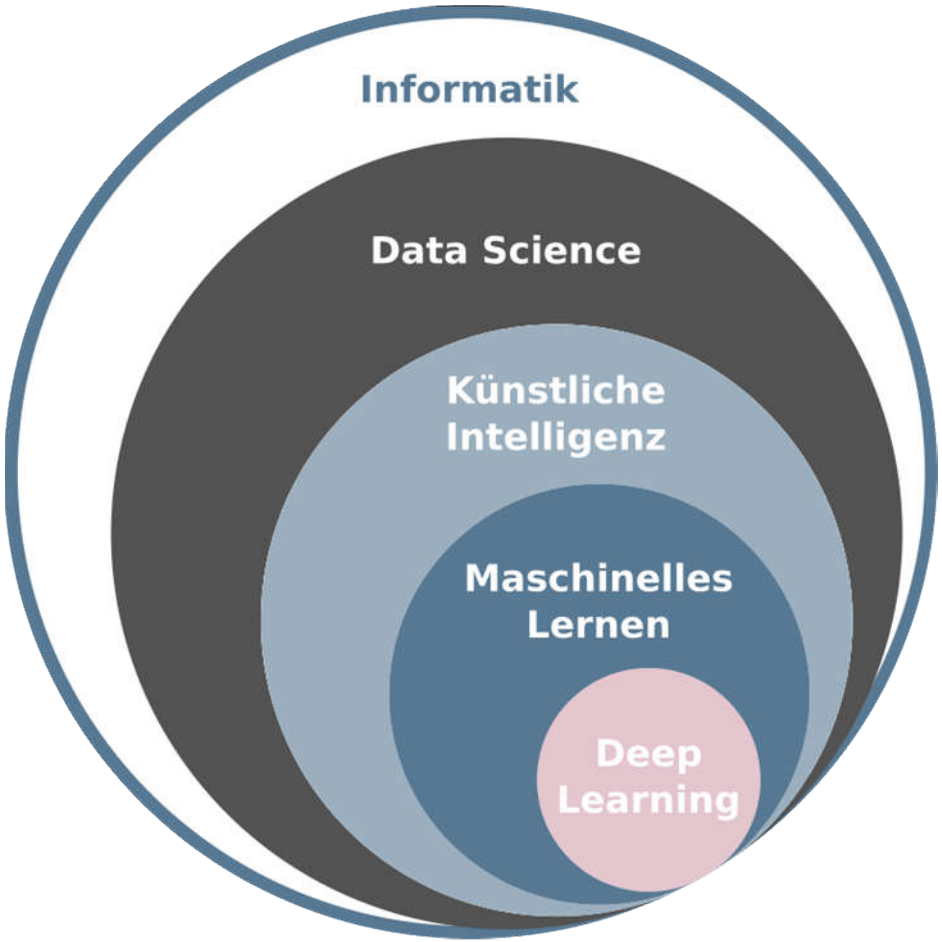
Einordnung
→ Künstliche Intelligenz



- **Künstliche Intelligenz** ist ein Fachgebiet der Informatik
- setzt intelligentes Verhalten in Algorithmen um
- (Ziel)
  - **automatisiert „menschenähnliche Intelligenz“ nachzubilden**.

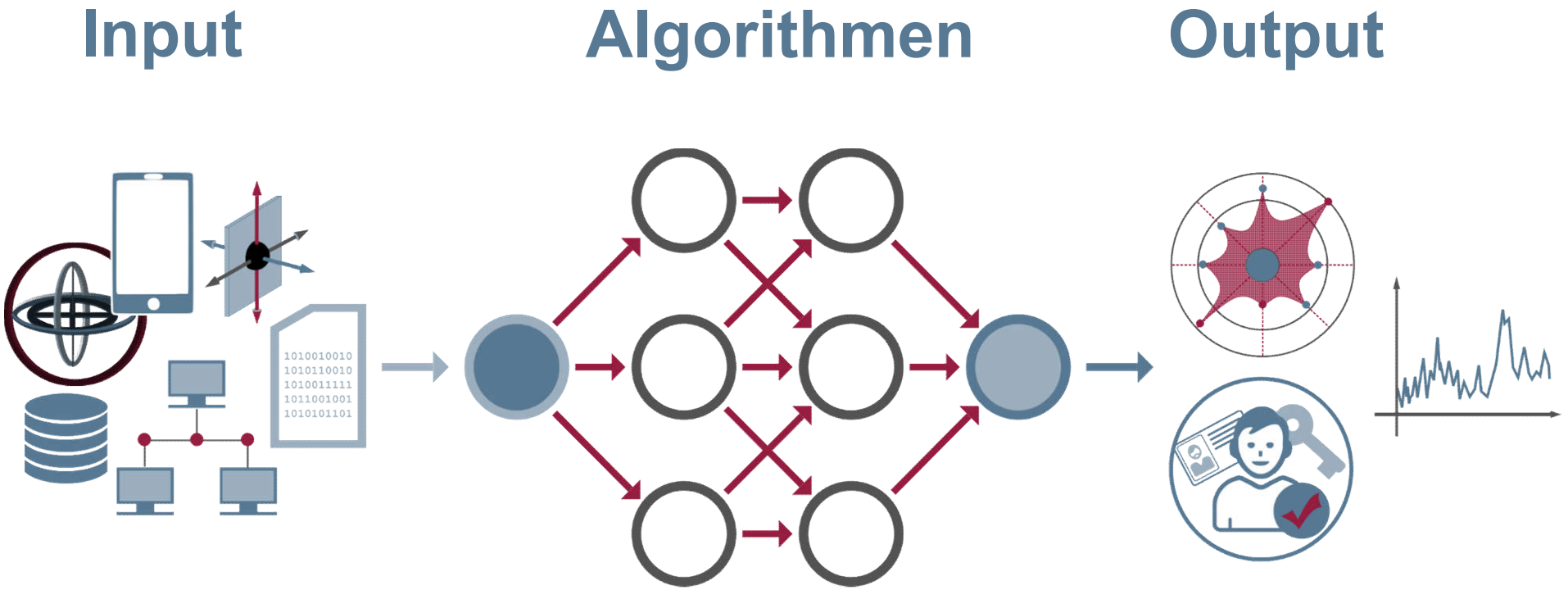


Einordnung
→ Maschinelles Lernen



- **Maschinelles Lernen** ist ein Begriff für die „künstliche“ **Generierung von Wissen aus Erfahrung** durch Computer.
- In **Lernphasen** lernen entsprechende ML-Algorithmen aus Beispielen **Muster und Gesetzmäßigkeiten**.
- Daraus erstehende Verallgemeinerungen können auf neue Daten angewendet werden.

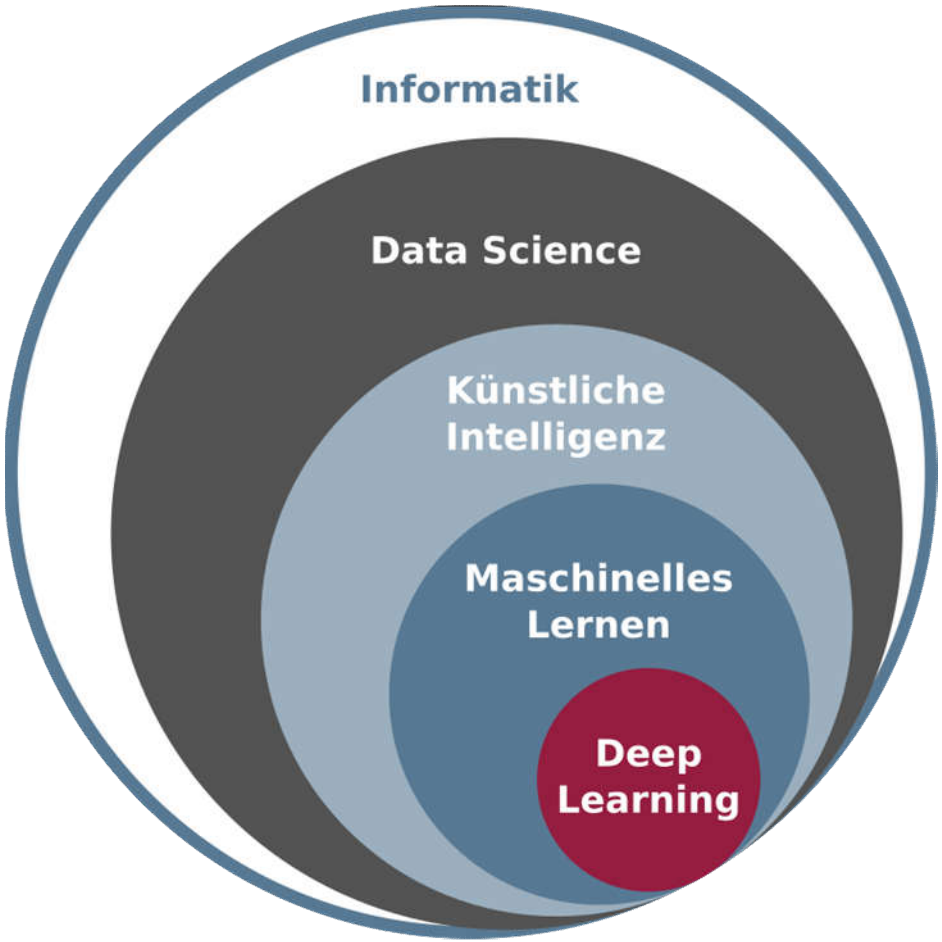
Maschinelles Lernen
→ Prinzip



Neuronales Netz:
Nachbildung der biologischen Struktur des Gehirns

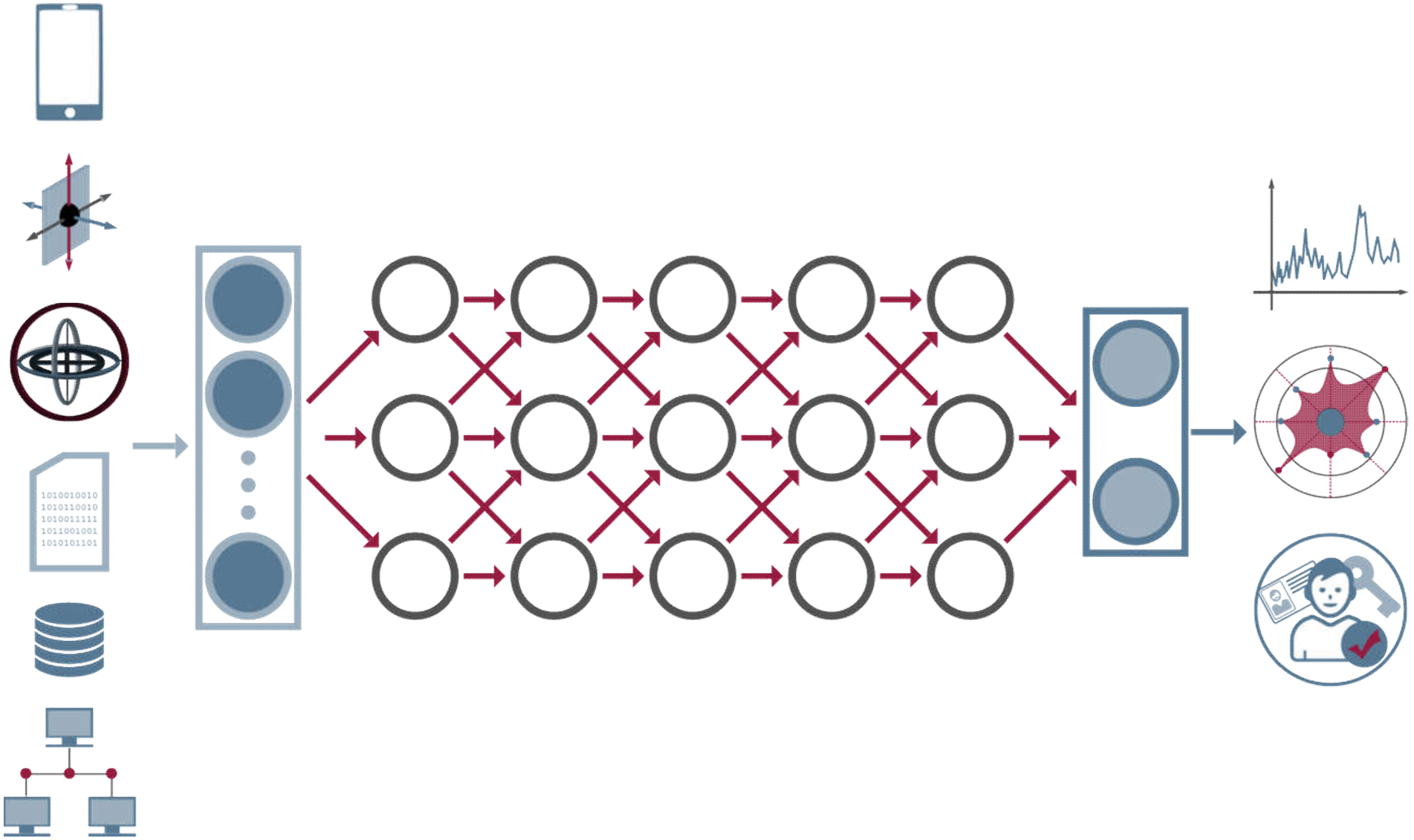
- Neuron = Kreis
- Synapse = Pfeil

Einordnung
→ Deep Learning



- Maschinelles Lernen wird noch effektiver durch:
  - **Deep Learning**
- Deep Learning ist eine Spezialisierung des maschinellen Lernens
- Kommt dem „menschlichen Gehirn“ am nächsten

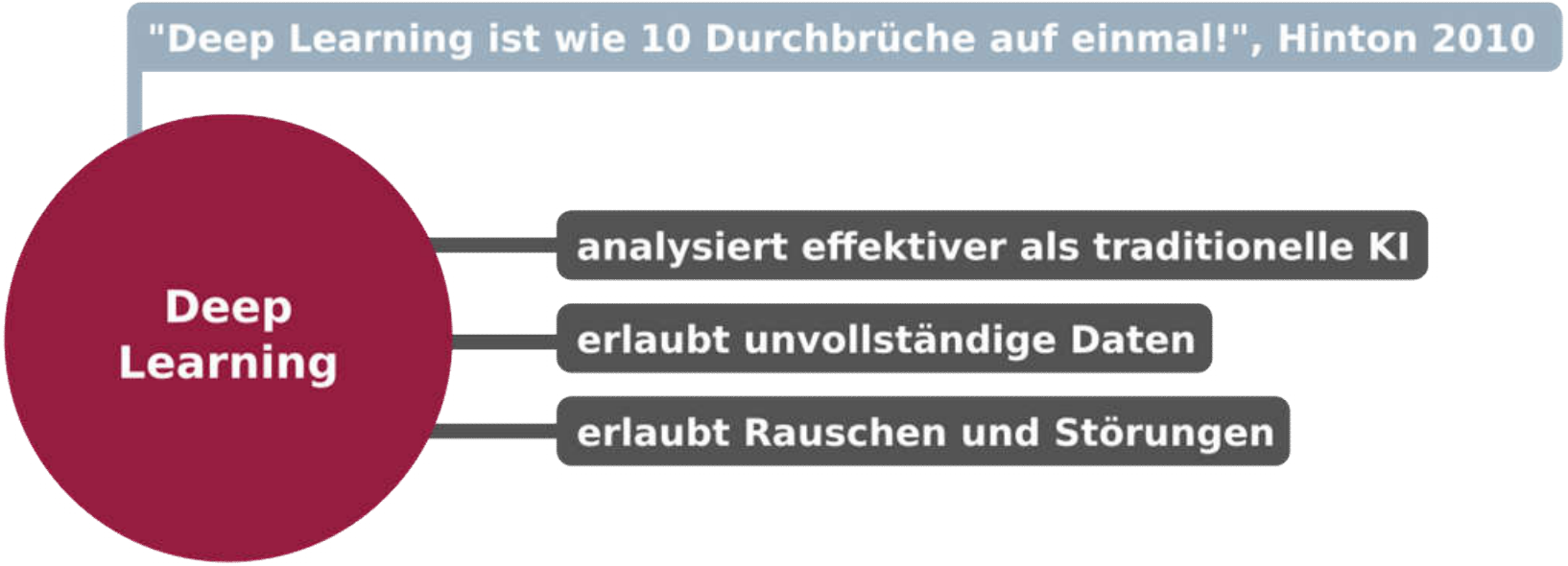
Deep Learning



Deep Learning
→ Der nächste Step

"Deep Learning ist wie 10 Durchbrüche auf einmal!", Hinton 2010

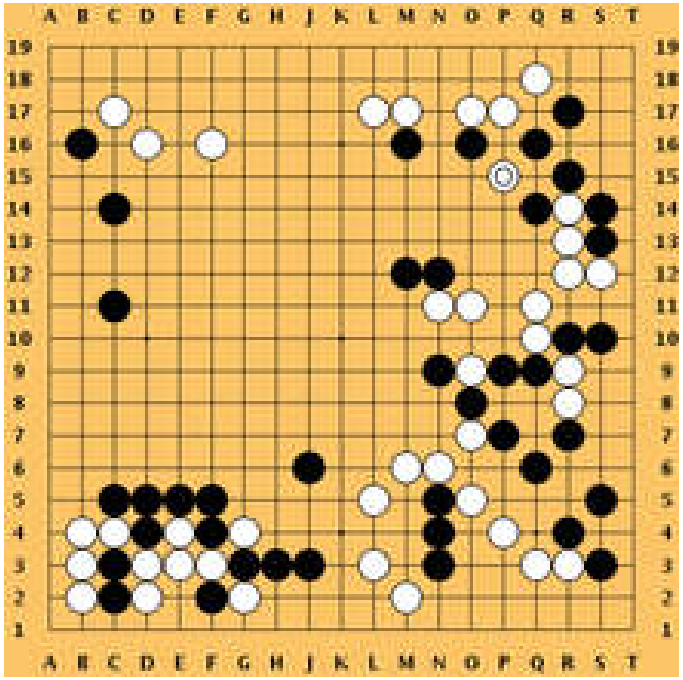
Deep Learning
→ Besonderheiten



Deep Learning
→ Anwender / Anwendungsfelder

Deep Learning
→ Informationen zu GO

- GO-Spielfeld 19x19 Felder
- 2 Spieler
- Regeln



- Möglichkeiten bei GO: 10^{170}
- Möglichkeiten beim Schach: 10^{43}

Deep Learning
→ AlphaGo

- Googles AlphaGo soll Menschen beim 3.000 Jahre alten Spiel GO besiegen

- Algorithmus: Monte Carlo Tree Search
- AlphaGo lernt die Regeln
- AlphaGo lernt aus vorhergegangenen Spielen

- 2015 Sieg über europäischen Meistern
- 2016 Sieg über den Weltmeister

Deep Learning
→ AlphaGo Zero (2017)

- Reduzierung des Einflusses des Monte Carlo Algorithmus
- Steigerung des Einflusses des **(deep)** neuronalen Netzes
- AlphaGo Zero lernt die Regeln
- AlphaGo Zero lernt **nicht** aus vorhergegangenen Spielen

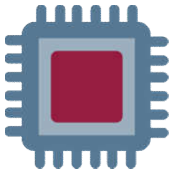
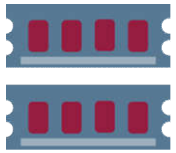
- Nach 3 Tagen Profi-Level
- Nach 21 Tagen AlphaGo Level
- Nach 40 Tagen weit darüber
- Nach 40 Tagen 3000 Jahre altes Wissen

Artificial Intelligence
→ Warum jetzt?

- Enorme Fortschritte der **Leistungsfähigkeit von Computern**
  - Zentrale Speicherung und Verarbeitung von massenhaften Input-Daten möglich

- Viele **umfangreiche Prozesse** von maschinellem Lernen in akzeptabler Zeit durchführbar
  - Parallelisierung steigert diese Leistung
  - Hohe Geschwindigkeiten in der Datenübertragung erlauben ein Auslagern verschiedener Prozesse auf weitere Server

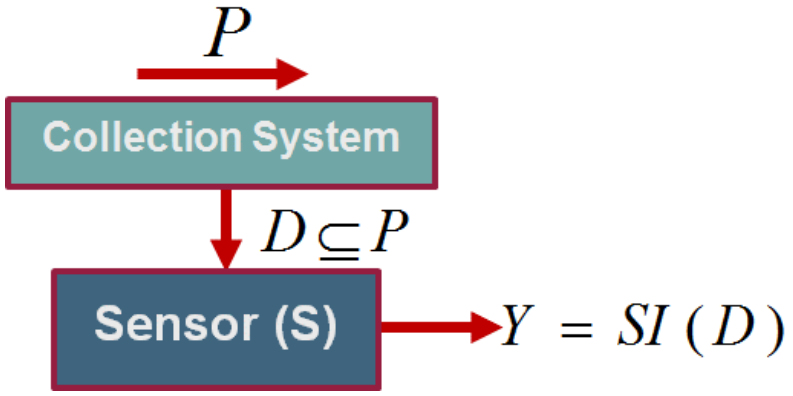
Entwicklungen der Leistungsfähigkeit
→ Hardware

	2008	2018
	Kerne: 2 Taktfrequenz: 1,6 GHz Cache: 4 MB	Kerne: 20 Taktfrequenz: 2,2 GHz Cache: 14 MB
	Arbeitsspeicher: 4 GB Speichertakt: 667 MHz	Arbeitsspeicher: 64 GB Speichertakt: 2.133 MHz
	Speicher: 584 GB (SCSI) Transferrate: 3 Gbps	Speicher: 960 GB (SSD) Transferrate: 6 Gbps
	Übertragungsrate: 100 Mbps	Übertragungsrate: 10 Gbps

Richtwert: 10.000 €
Stand: Februar 2018

Entwicklungen der Leistungsfähigkeit → Immer mehr Input-Daten

- **Quantität gestiegen**
  - Immer mehr Daten durch die Digitalisierung
  - Sehr viele Sensoren (in Diensten, am Körper, im Auto, …)
- **Qualität gestiegen**
  - Feingraduale Aggregation möglich
  - Weitere Individualisierung der (persönlichen) Daten (Smartphone, Smartwatch, …)
  - Sicherheitsrelevante Informationen (SI) sammeln



$$SI(Y) \leq SI(D) \leq SI(P)$$

ideal: ***SI (Y) = SI (P)***

Entwicklungen der Leistungsfähigkeit
→ Immer bessere ML-Algorithmen

- Gesamtablauf wird optimiert
  - Reduktion der Komplexität durch intelligent gewählte Input-Daten
  - Rechenaufwand wird minimiert durch effizientere ML-Algorithmen

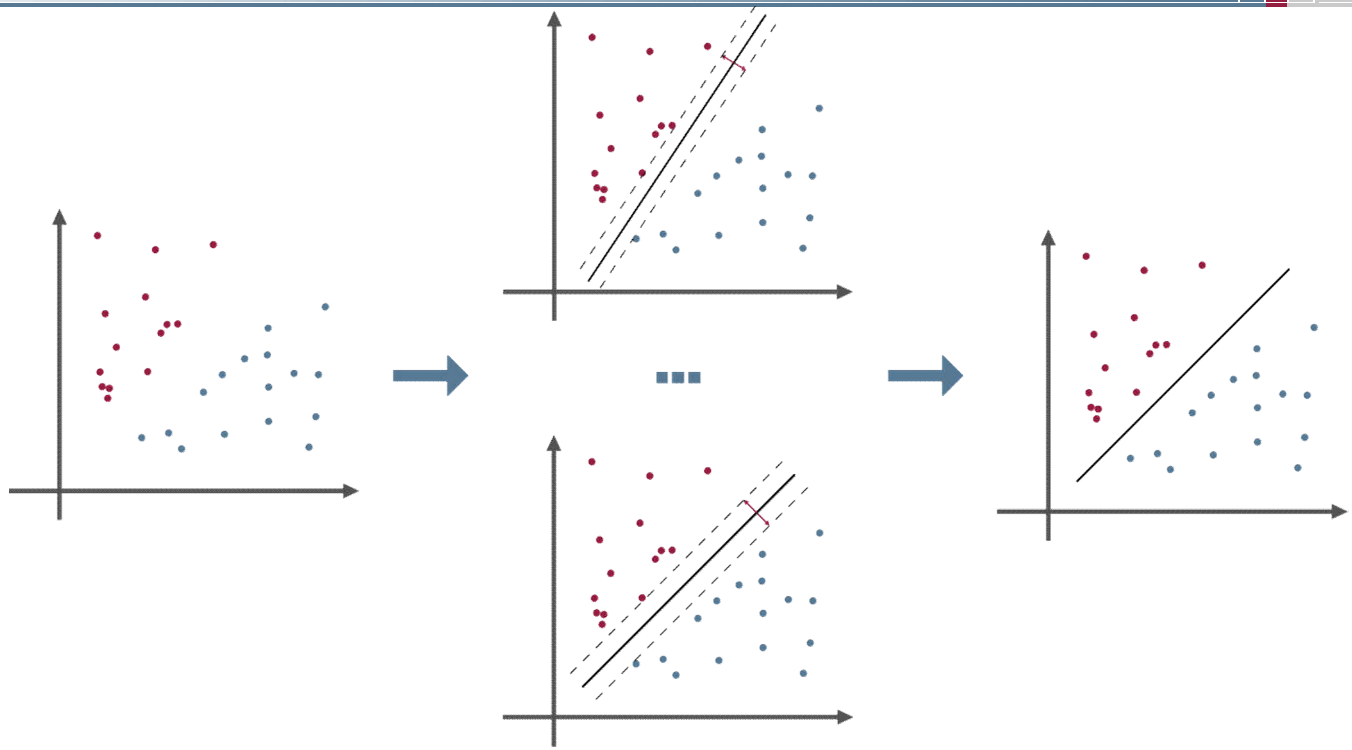
Algorithmen des maschinellen Lernens werden erst durch diese Verbesserungen praktisch umsetzbar!

ML-Algorithmus
→ überwachtes Lernen

- Ziele des überwachten Lernens
  - **Regression:** Vorhersagen von numerischen Werten
  - **Klassifizierung:** Einteilung von Daten in Klassen
- Beispiele
  - Vorhersage des Gehalts einer Person
  - Erkennung einer Blütenart anhand der äußeren Merkmale
- Eingabedaten enthalten erwartete Ergebnisse
- Einteilung der Daten in Trainings- und Testmengen

- **ML-Algorithmus**
  - Support-Vector-Machine (SVM)
  - k-Nearest-Neighbour (kNN)

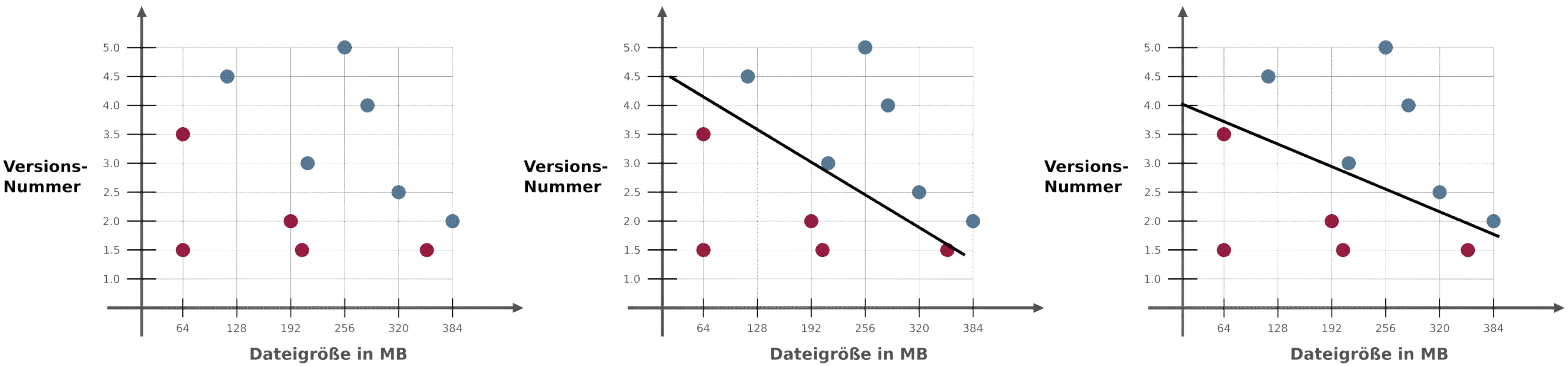
ML-Algorithmus
→ Support-Vector-Machine(SVM) - Training



- **Input-Daten:**
  - bereits klassifizierte Daten
  - Abstandsmaß
- **ML-Algorithmus:**
  - ermitteln von Geraden zur Trennung der Daten
  - Bewertung durch Abstand zu den Punkten
  - Wahl der Geraden mit maximalem Abstand zu beiden Klassen
- **Output:**

  Gerade als Modell zur Klassifizierung

ML-Algorithmus
→ SVM - Beispiel Training



Dateigröße (MB)	64	64	384	320	192	198	120	256	360	200	270
Versions-Nr.	1,5	3,5	2	2,5	2	1,5	4,5	5	1,5	3	4
Malware	ja	ja	nein	nein	ja	ja	nein	nein	ja	nein	nein

- **Input-Daten:**

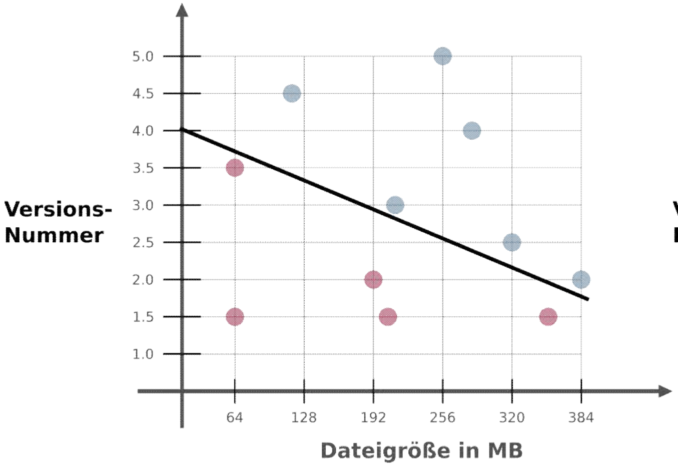
  Programmdateien mit entsprechender Klassifikation

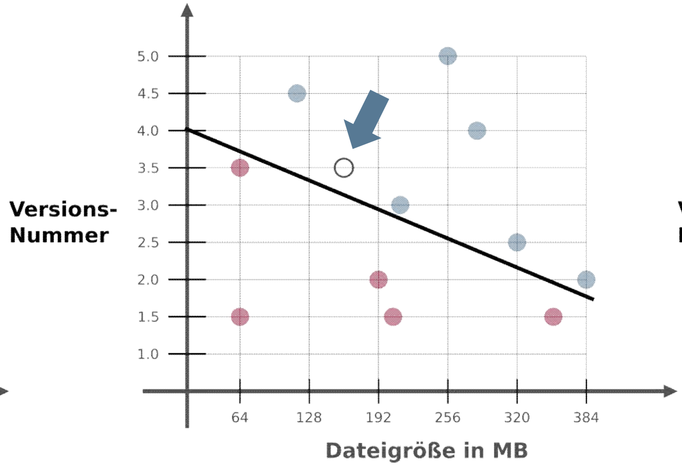
- **ML-Algorithmus:**
  - Ermittlung der Geraden, welche die Daten trennen
  - Wahl der Geraden

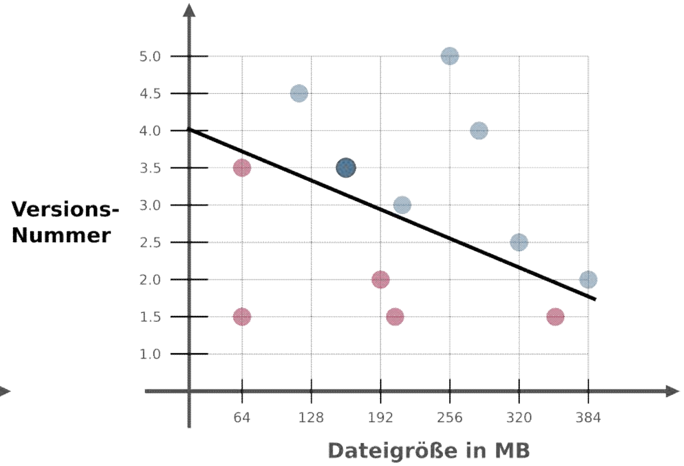
- **Output:**

  Gerade als Modell zur Klassifizierung von Programmdateien als schadhaft / harmlos

ML-Algorithmus
→ SVM - Beispiel Anwendung







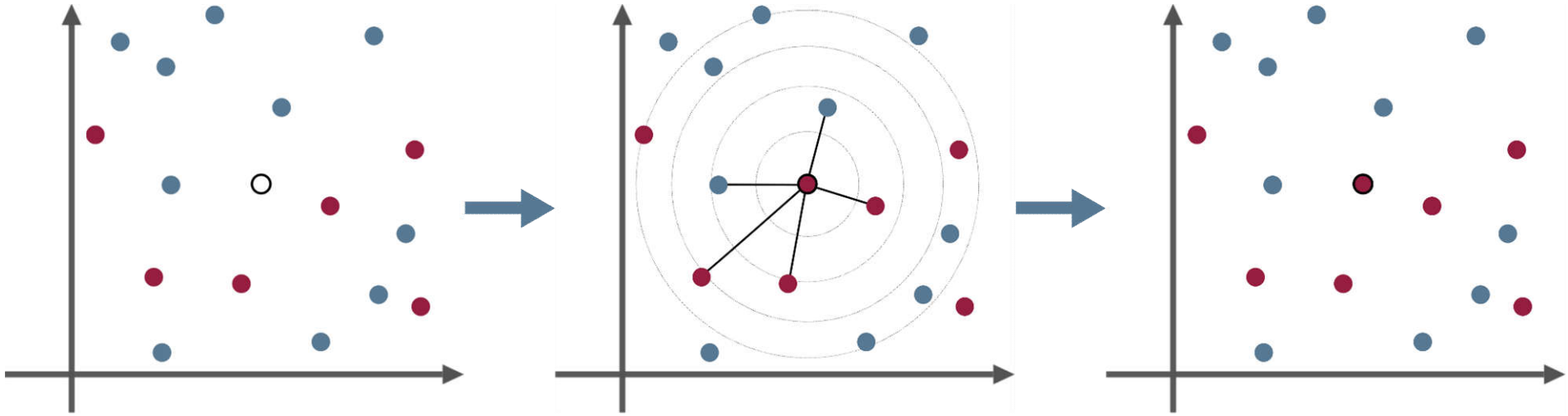
Dateigröße (MB)	64	64	384	320	192	198	120	256	360	200	270	**160**
Versions-Nr.	1,5	3,5	2	2,5	2	1,5	4,5	5	1,5	3	4	**3,5**
Malware	ja	ja	nein	nein	ja	ja	nein	nein	ja	nein	nein	**?**

- **Input-Daten:**
  - Modell zur Unterscheidung von Versionen mit und ohne Schadfunktion
  - zu beurteilende Programmdatei
- **ML-Algorithmus:**

  Berechnung der Lage der Programmdatei
- **Output:**

  Lage der Punkte zum Modell klassifiziert die Programmdatei als **harmlos**

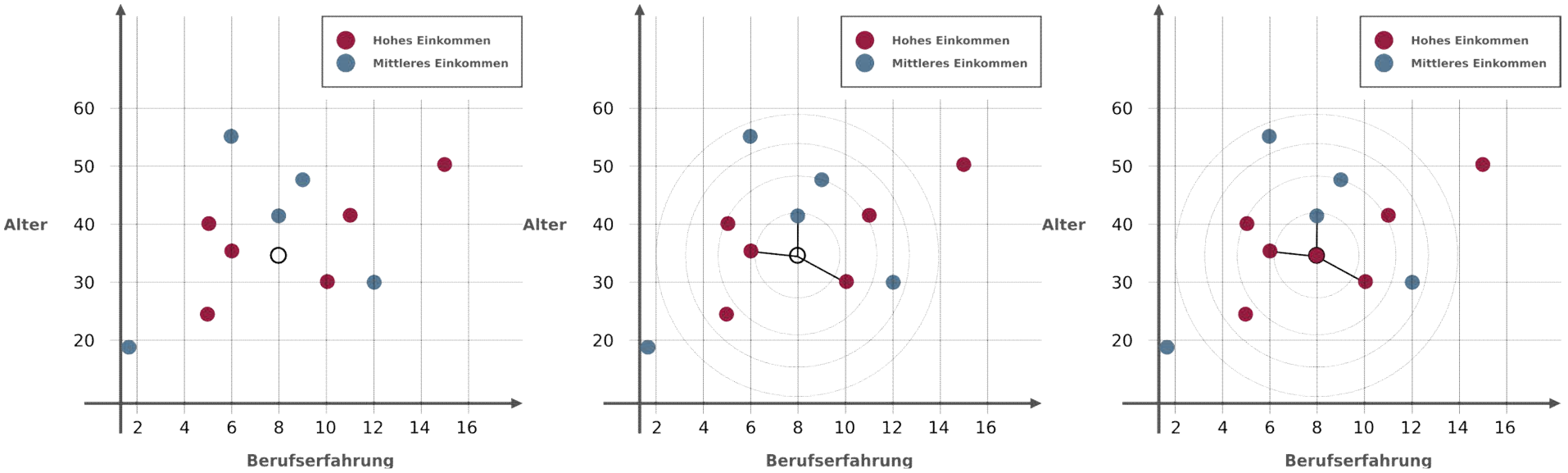
ML-Algorithmus
→ k-Nearest-Neighbour (kNN)



- **Input-Daten:**
  - Bereits klassifizierte Objekte
  - unklassifiziertes Objekt
  - Anzahl der zu betrachtenden Nachbarobjekte k
- **ML-Algorithmus:**
  - Berechnung der Distanz zu allen anderen Objekten
  - Betrachtung der k nächsten Nachbarobjekte
  - Zuordnung zur am häufigsten vorkommenden Klasse
- **Output:**

  Klassifizierung des neuen Objekts

ML-Algorithmus
→ k-Nearest-Neighbour - Beispiel



Berufs-erfahrung	15	5	12	6	8	5	1	9	10	11	11	6	**8**
Alter	50	40	30	36	42	25	19	48	30	27	42	55	**35**
Gehalt	hoch	hoch	mittel	hoch	mittel	hoch	mittel	mittel	hoch	hoch	hoch	mittel	**?**

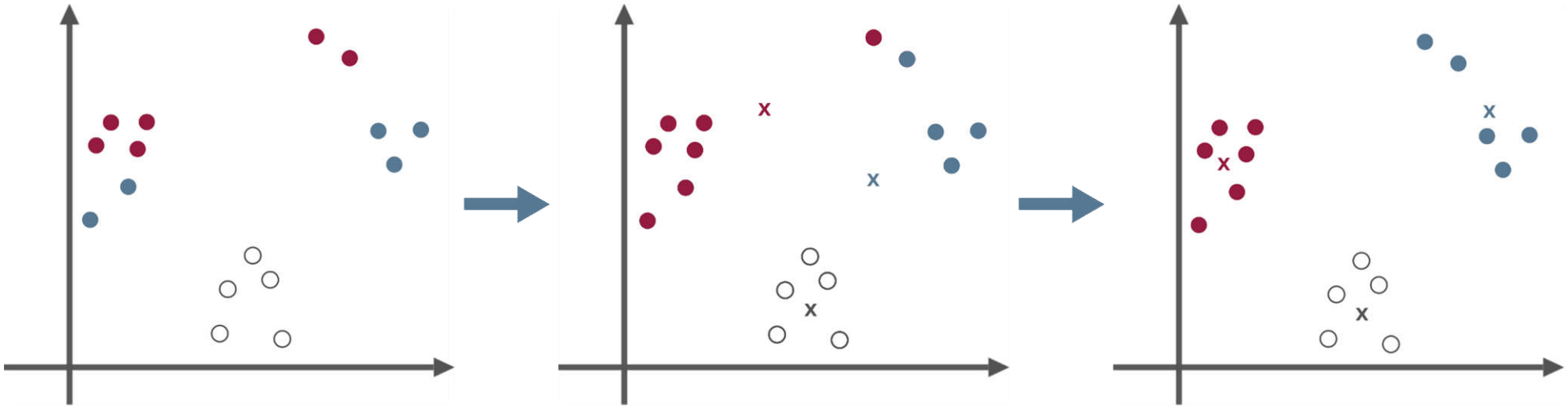
- **Input-Daten:**
  - Gehaltsdaten
  - Daten des neuen Mitarbeiters
- **ML-Algorithmus:**
  - Berechnung der Distanzen
  - Betrachtung der 3 nächsten Mitarbeiter
- **Output:**

  Mitarbeiter kann ein hohes Gehalt erwarten

ML-Algorithmus
→ unüberwachtes Lernen

- **Stärke im Suchen nach Mustern in unklassifizierten Daten**
- Erwartungshaltung an diesen Ansatz:
  - Muster erkennen, die vorher anders nicht greifbar waren
- ML-Algorithmus lernt selbstständig
- Klassische Fehler werden in diesem Sinne nicht produziert
- **ML-Algorithmus**
  - Clustering setzt ähnliche Datengruppen miteinander in Verbindung
    - k-Means-Algorithmus
    - Hierarchische Clustering-Verfahren
- **Problem:** Lernt der ML-Algorithmus in die gewünschte Richtung?

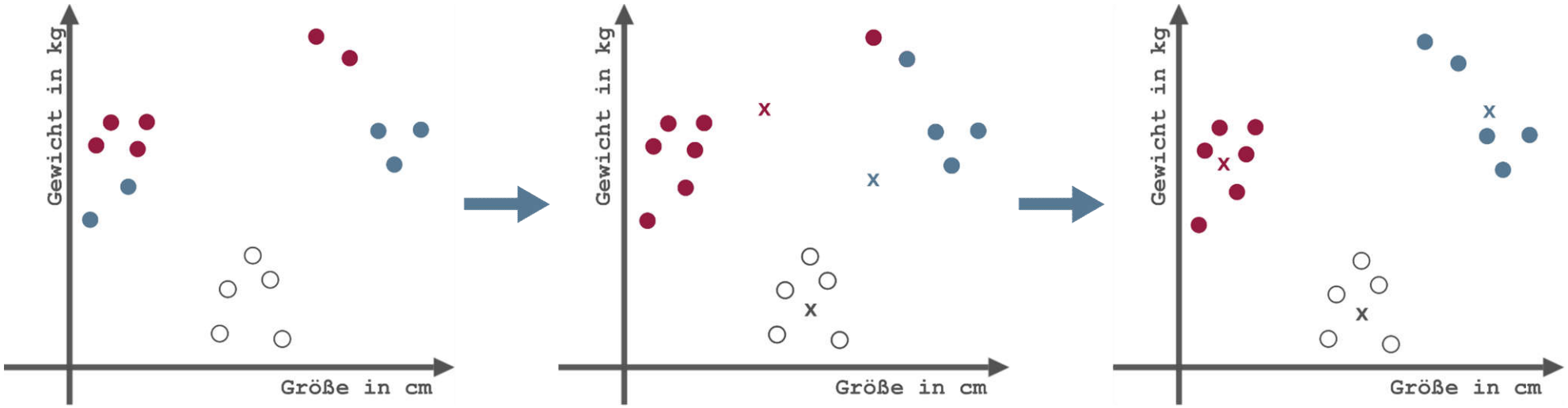
ML-Algorithmus
→ k-Means-Algorithmus



- **Input-Daten:**
  - beliebige Daten
  - Abstandsmaß
  - Anzahl k Cluster
  - Initiale Zuordnung der Elemente zu Clustern (z.B. zufällig)
- **ML-Algorithmus:**
  - Berechnung der **Schwerpunkte** (Zentroide)
  - Zuordnung der Elemente zu Cluster mit dem nächsten Zentroid
  - Neuberechnung der Zentroide und erneute Zuordnung
- **Output:**

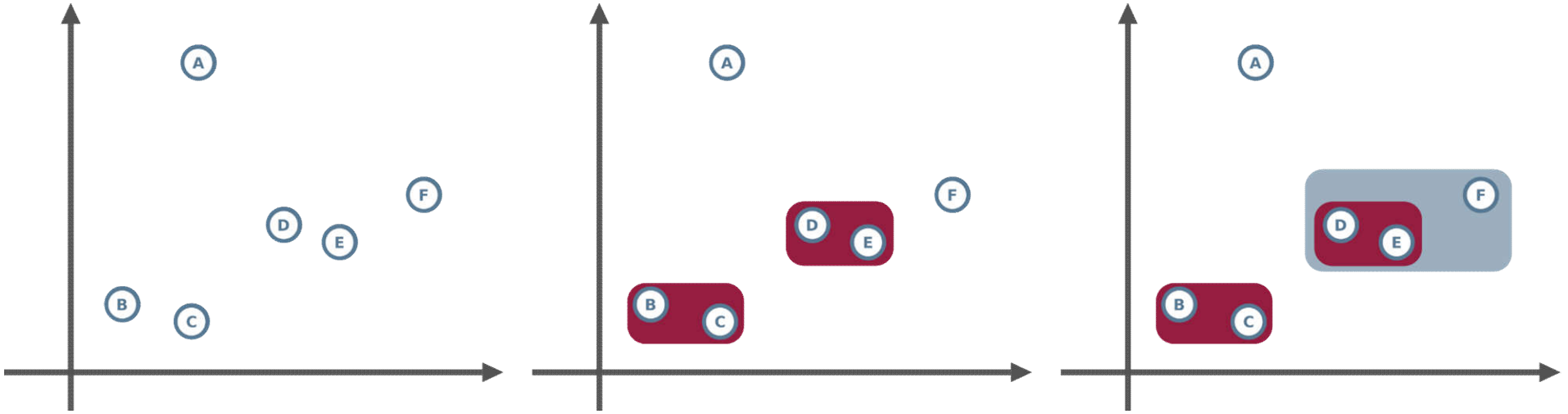
  Einteilung der Objekte in k Cluster

ML-Algorithmus
→ k-Means-Algorithmus - Beispiel



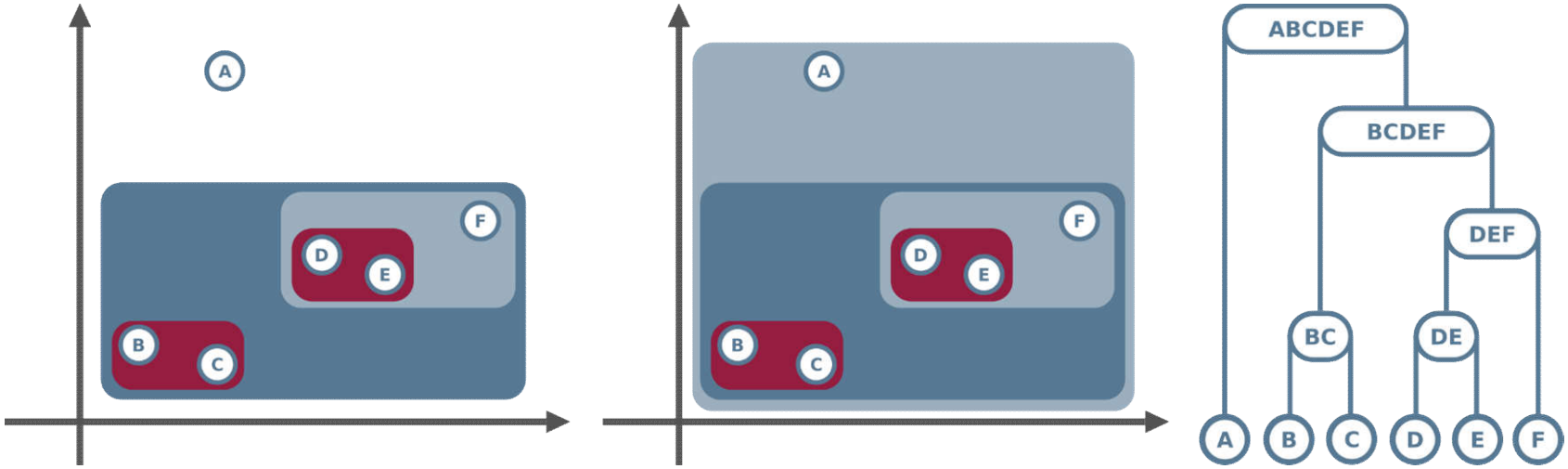
- **Input-Daten:**
  - Daten von Meeresschildkröten, Gazellen, Pferde, …
  - Abstandsmaß
  - k = 3
  - Initiale Zuordnung nach Gewicht, Größe
- **ML-Algorithmus:**
  - Berechnung der Durchschnitte
  - Zuordnung der Elemente zur Tierart mit dem nächsten Zentroid
  - Neuberechnung der Zentroide und erneute Zuordnung
- **Output:**
  - Einteilung der Tiere in die drei Tierarten
    - Rot = Meeres-schildkröten
    - Weiß = Gazellen
    - Blau = Pferde

ML-Algorithmus
→ Hierarchische Clustering-Verfahren



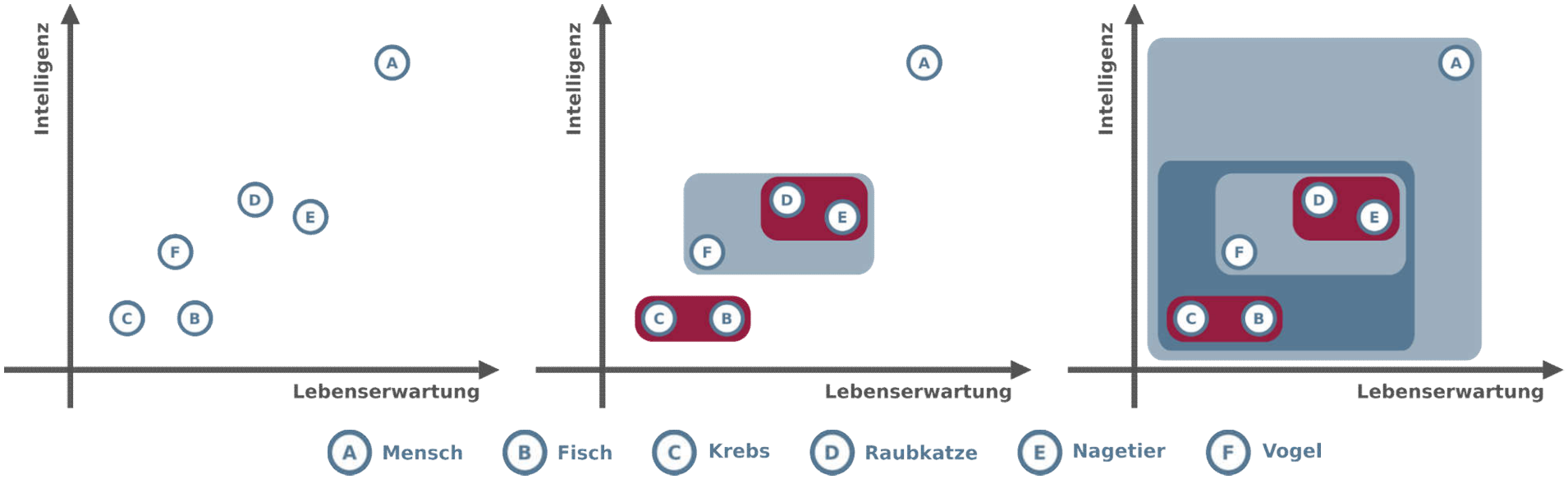
- **Input-Daten:**
  - beliebige Daten
  - Ähnlichkeitsmaß
- **ML-Algorithmus:**
  - jeder Datenpunkt ist ein eigenes Cluster
  - ähnlichste Cluster werden zuerst zusammengeführt
  - entstandene Cluster werden erneut als Eingabedaten verwendet
  - iteratives Zusammenführen der Cluster induziert eine hierarchische Struktur

ML-Algorithmus
→ Hierarchische Clustering-Verfahren



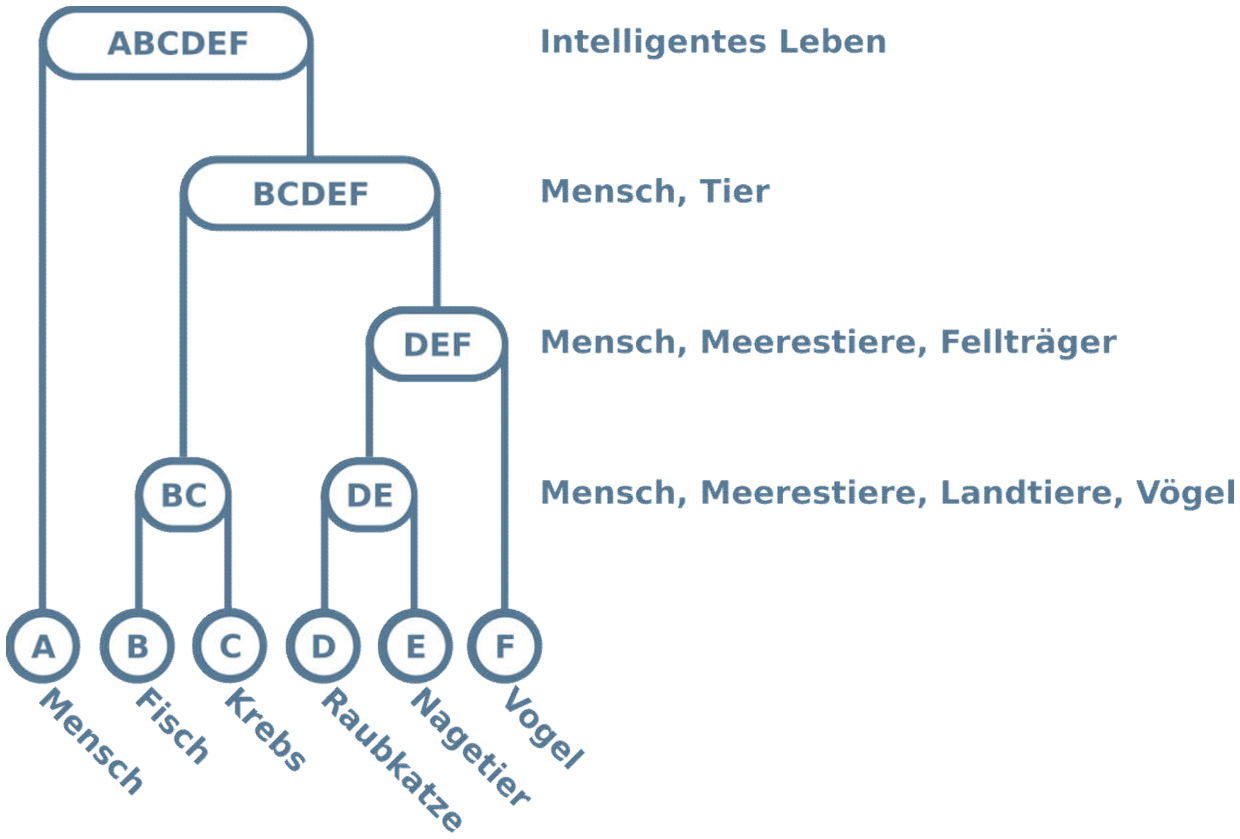
- **Output:**
  - Hierarchische Beziehungen zueinander in Form eines Binärbaums (Dendrogramm)

ML-Algorithmus
→ Hierarchische Clustering-Verfahren - Beispiel



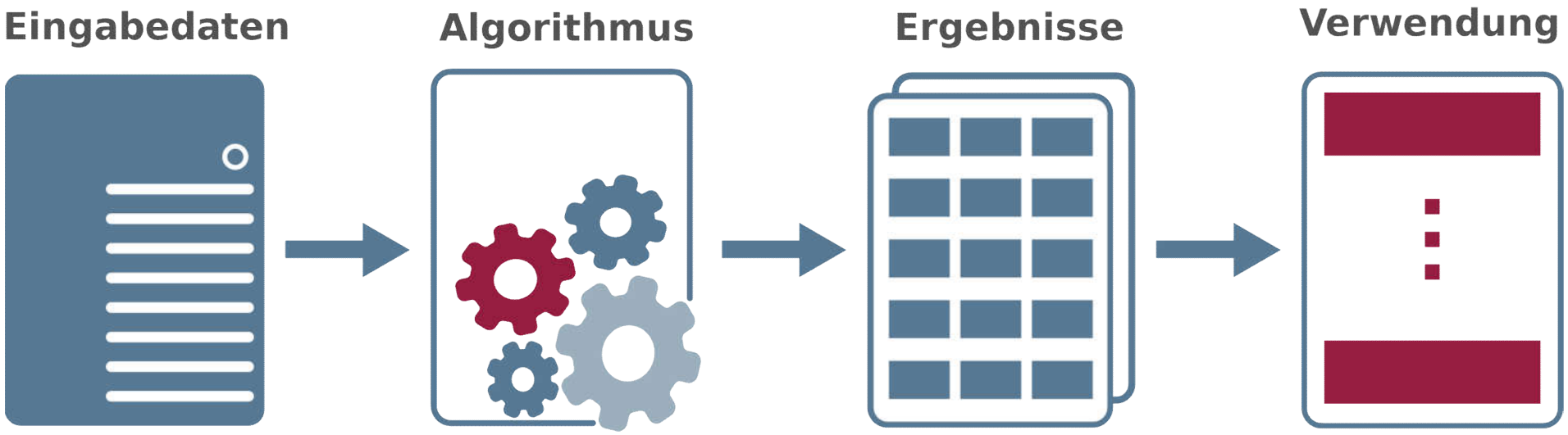
- **Input-Daten:**
  - Daten von Lebewesen
  - Ähnlichkeitsmaß
- **ML-Algorithmus:**
  - Lebewesen stellen eigene Cluster dar
  - ähnlichste Lebewesen werden zuerst zusammengefasst
    - Raubkatzen und Nagetiere zu Landtieren
    - Fisch und Krebs zu Meerestieren
  - Welche Gruppe passt eher zu den Vögeln?

ML-Algorithmus
→ Hierarchische Clustering-Verfahren - Beispiel



- **Output:**
  - Einordnung der Lebewesen in hierarchische Gruppierungen

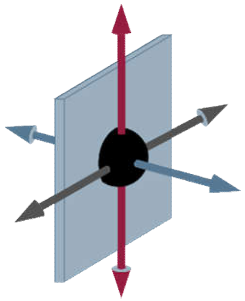
Maschinelles Lernen
→ Workflow

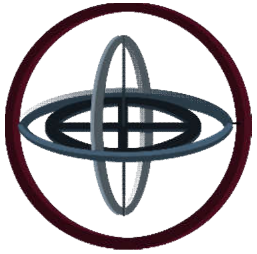


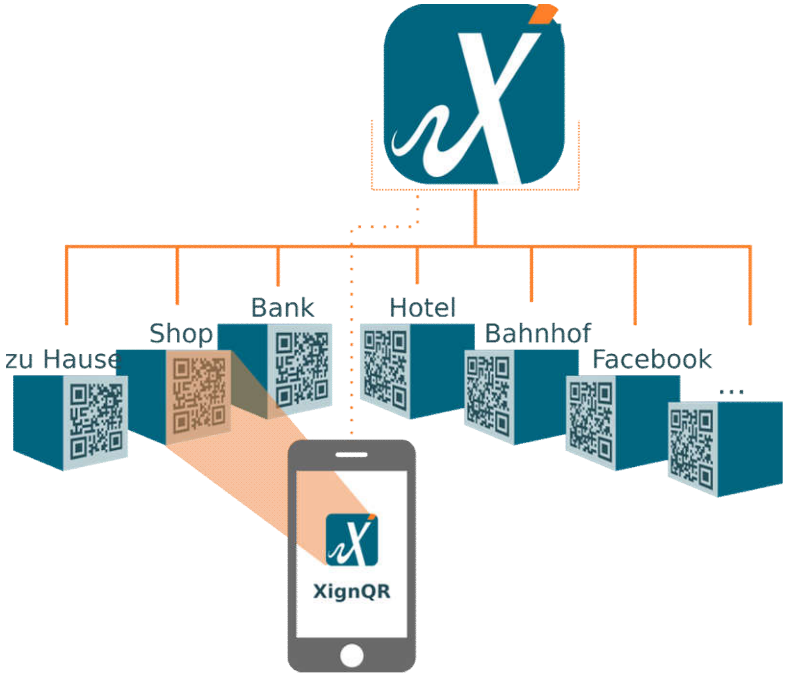
- Quellen der Eingabedaten auswählen
- Eingabedaten aufbereiten
  - Offensichtliche Fehler beseitigen
  - Strategien für fehlende Datenanteile anwenden
  - Normalisierung der Daten
  - Anwendung von Strategien zur Reduktion der Daten-Dimension (je nach Domäne)
- Algorithmus anwenden
- Ergebnis interpretieren
- Ergebnisse verwenden

Anwendungsbeispiel „XignQR“
→ Verhaltensmustererkennung

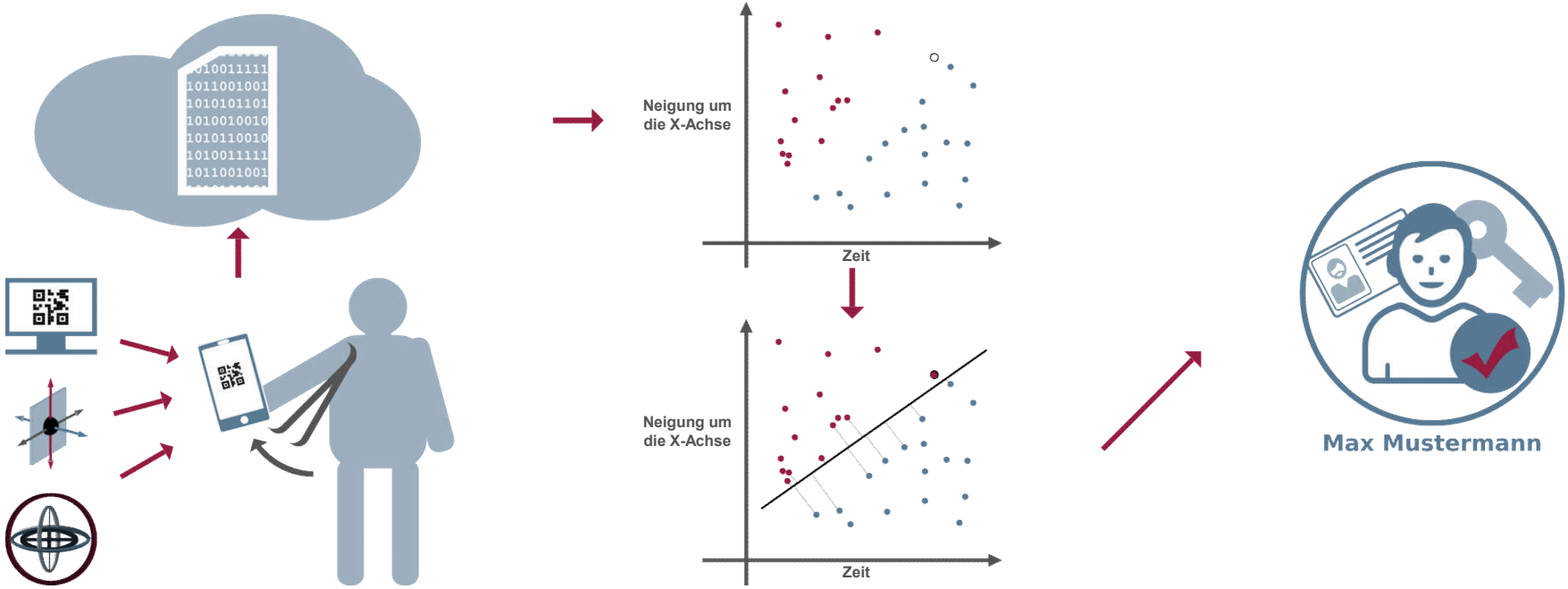
- Ein Nutzer wird automatisiert an der Art und Weise der Nutzung beim QR-Code Scannen erkannt.
- Während das gesamten Vorgangs werden passive biometrische Bewegungsdaten erfasst.
- Datenerfassung durch
  - **Beschleunigungssensor**
  - **Lagesensor**





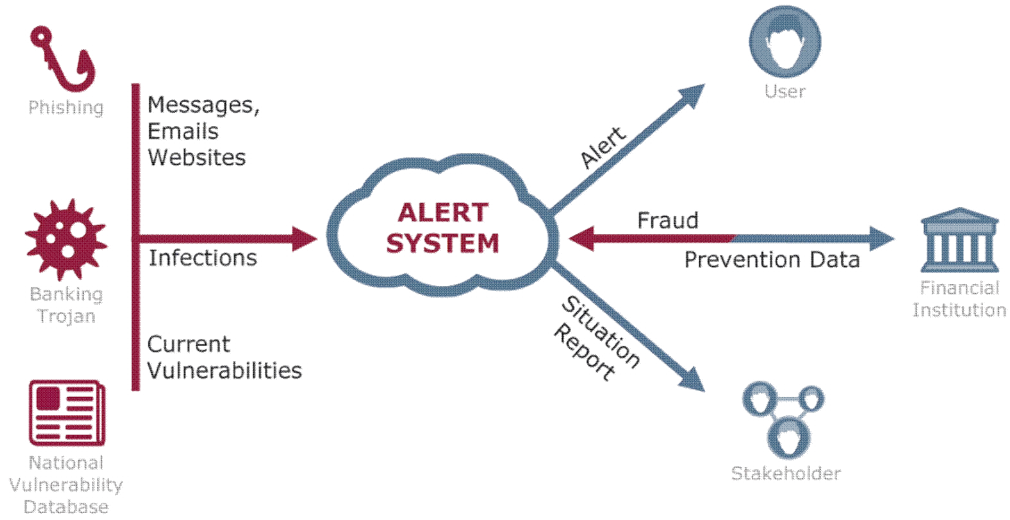


Anwendungsbeispiel „XignQR“



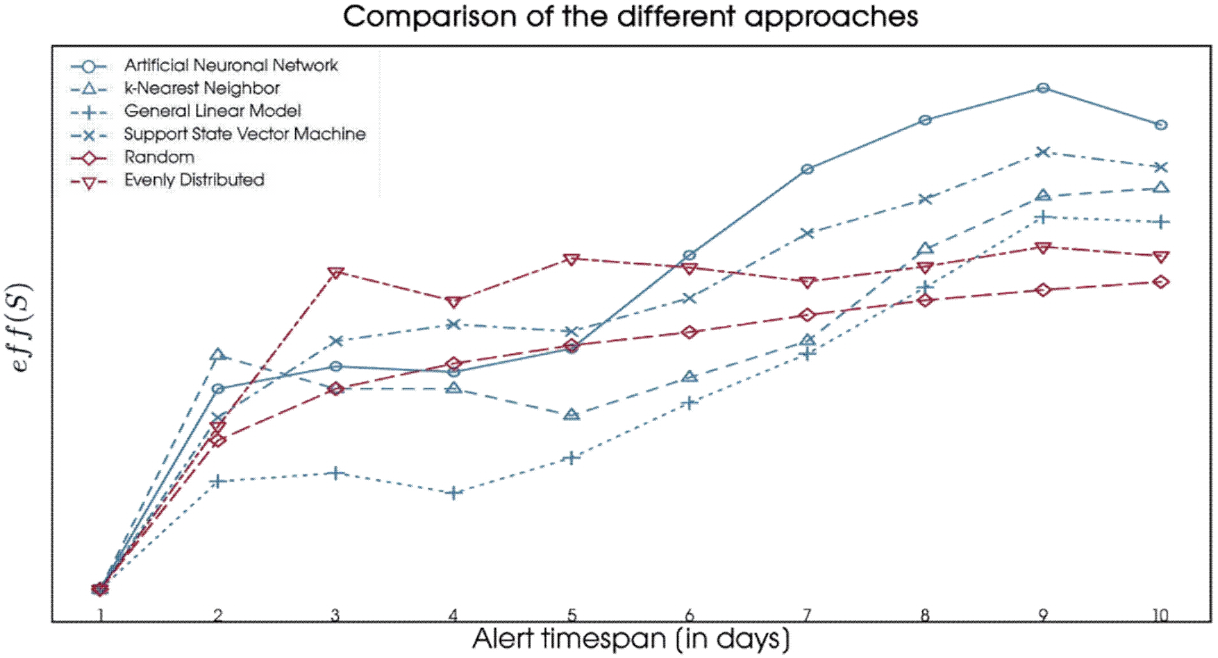
- **Input-Daten:**
  - Nutzer holt Gerät aus Hosentasche
  - Erfassen von **Lage** und **Beschleunigung** des Smartphones
- **ML-Algorithmus:**
  - Daten werden strukturiert
  - rote Übereinstimmung ist positive Klassifizierung
  - blau eine negative Klassifizierung (bspw. anderer Nutzer)
- **Output:**
  - Authentisierung ist entweder erfolgreich oder schlägt fehl

Anwendungsbeispiel
→ Betrugsschutz im Online-Banking (BOB)



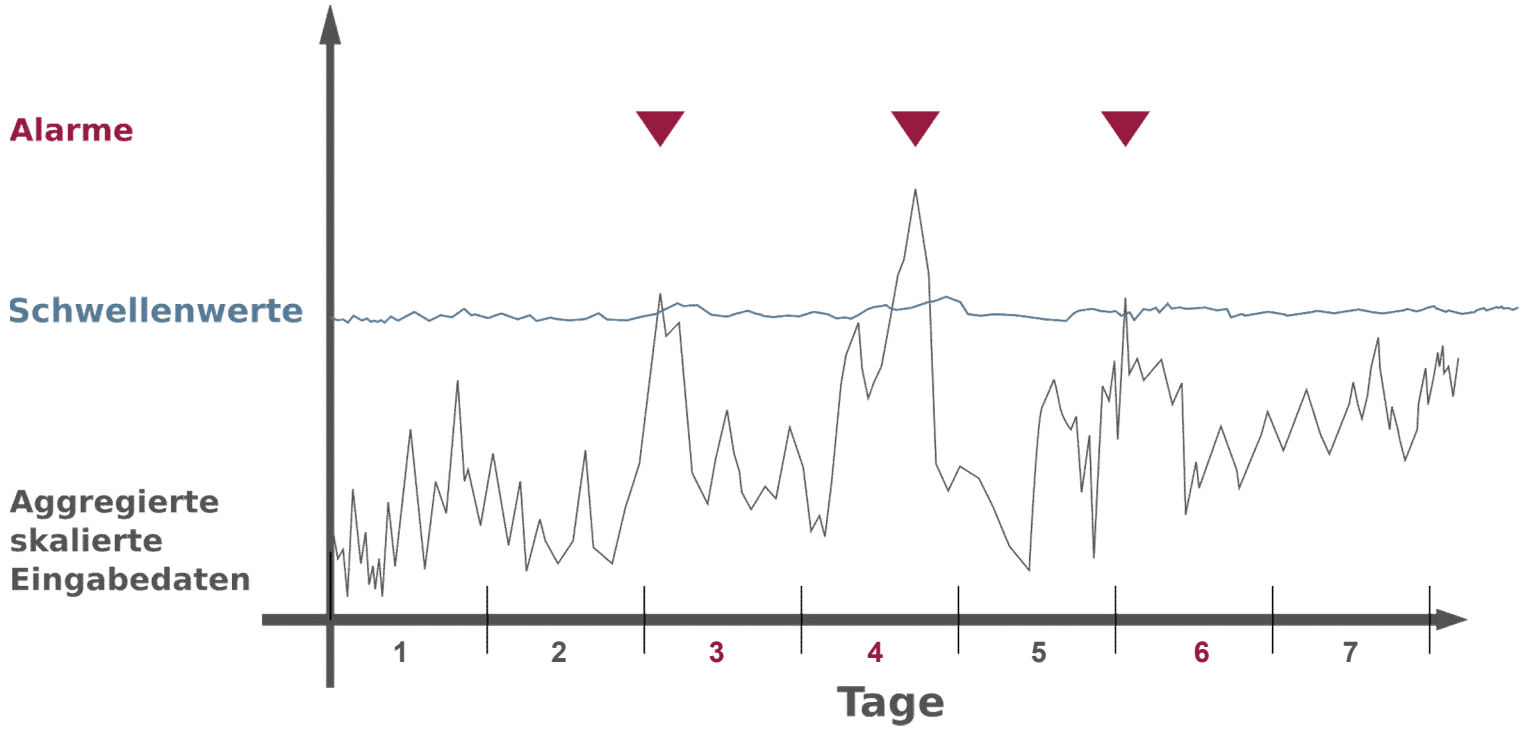
- Phishing-Seiten, Banking-Trojaner, Software-Schwachstellen und andere Quellen als Eingabedaten für die Identifizierung eines erhöhten Bedrohungslevels
- Verarbeitung durch das BOB-Alert-System
  - kNN, SVM, neuronale Netze
- Intelligente Generierung von Alarm-Meldungen mit Hilfe verschiedenster ML-Algorithmus

Anwendungsbeispiel
→ Betrugsschutz im Online-Banking (BOB)



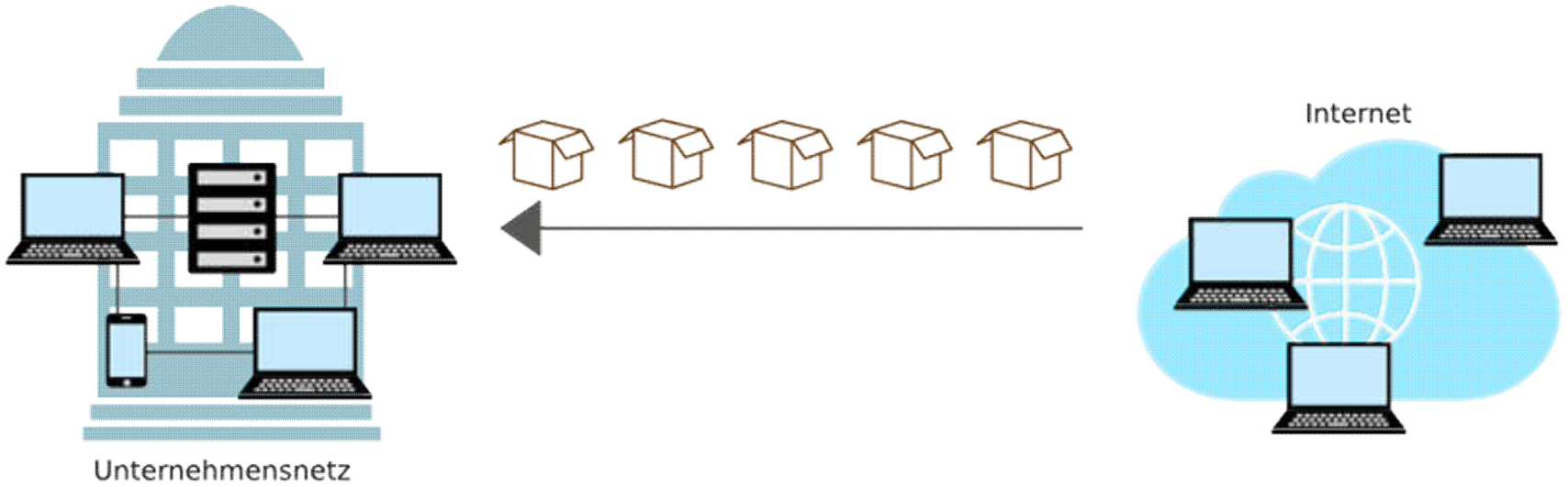
- **Input-Daten:**
  - Erhöhtes auftreten Banking-Trojaner „Zeus“
    - Familie
    - Datum
    - Anzahl der Auftretens
- **ML-Algorithmus:**
  - Daten werden strukturiert, verarbeitet und analysiert
  - erzeugt numerischen Bedrohungswert
- **Output:**
  - Bedrohungswert
  - Alarm wegen Zeus-Kampagne

Alert-System
→ Ergebnis



- **Output:**
  - Vorhergesagte Bedrohungswerte überschreiten an den Tagen 3, 4 und 6 den für dieses System eingestellten Schwellenwert
  - da Schwellenwert überschritten wurde wird ein Alarm ausgelöst

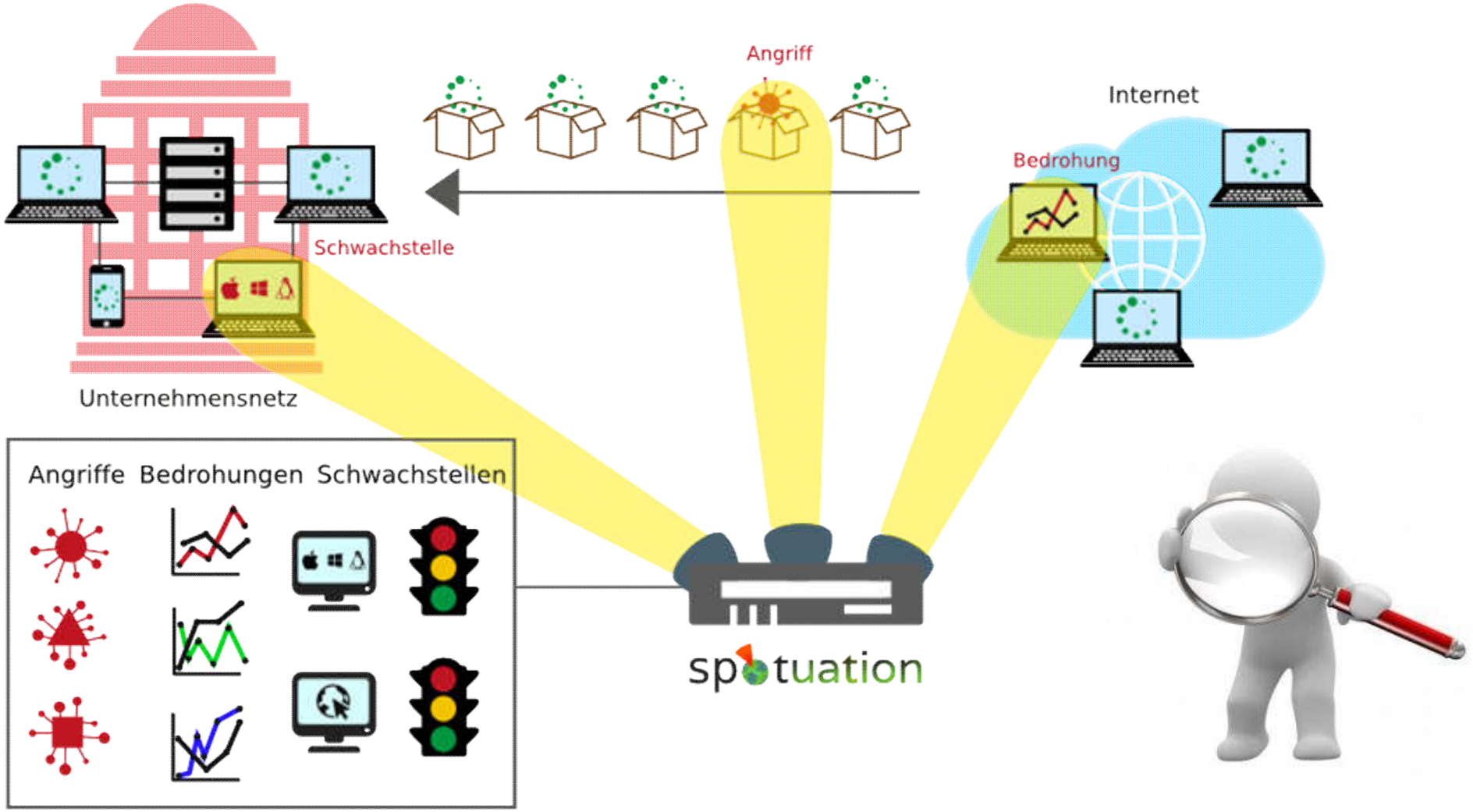
Anwendungsbeispiel „Spotuation“
→ Idee



- Wer greift auf das Netzwerk zu?
- Wie sieht das Netzwerk aus?
- Welche Gefahren gibt es?
- Welche Schwachstellen sind vorhanden?

Anwendungsbeispiel „Spotuation“
→ Kommunikationslagebild

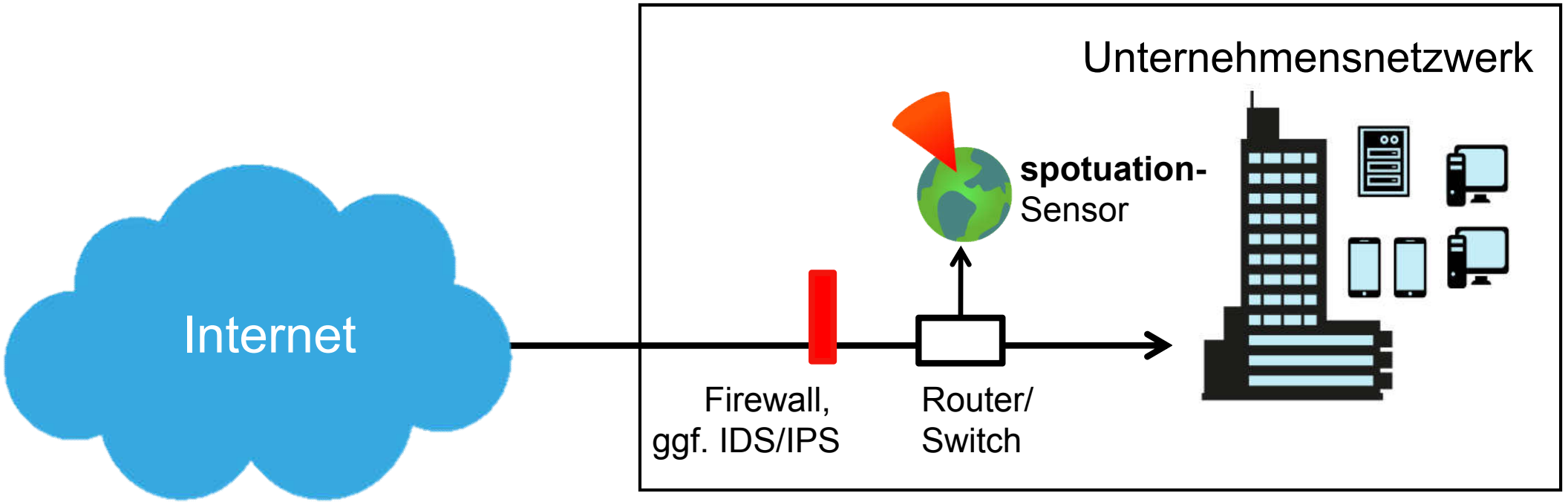
Angriffe, Bedrohungen und Schwachstellen im Überblick.



➢ **Das Kommunikationslagebild zeigt auch, was sicher ist!**

Anwendungsbeispiel „Spotuation“
→ Einfache Integration

- Integration des spotuation-Sensors:
  - Mittels Tap am Router oder per Mirror-/SPAN-Port im Switch/Router
- Konfiguration und Auswertung bequem per Browser

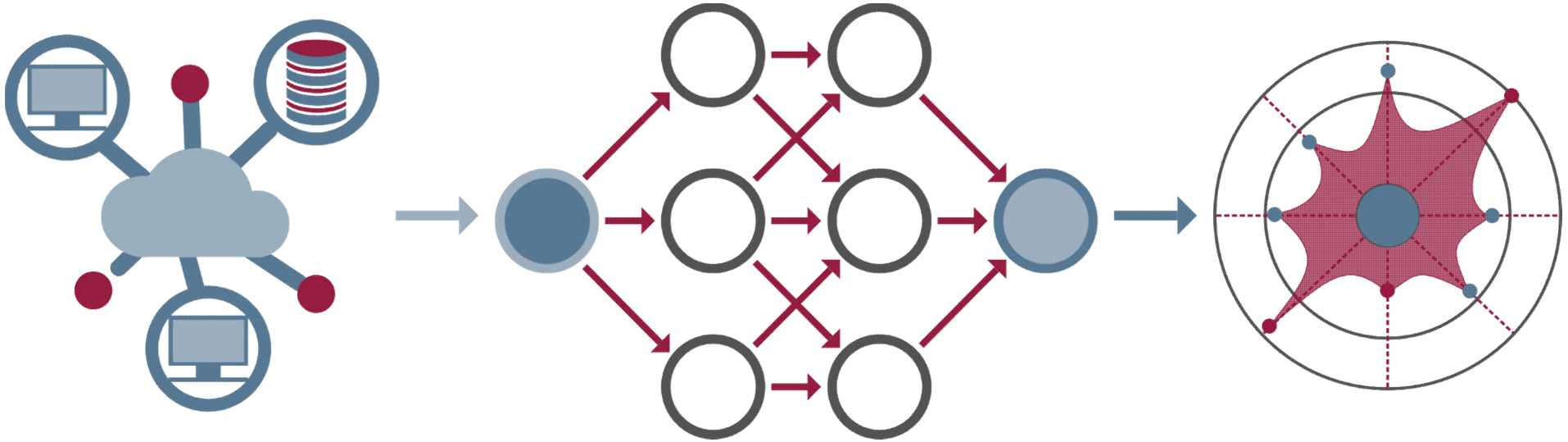


Anwendungsbeispiel „Spotuation“
→ Umsetzung

Kommunikations-parameter

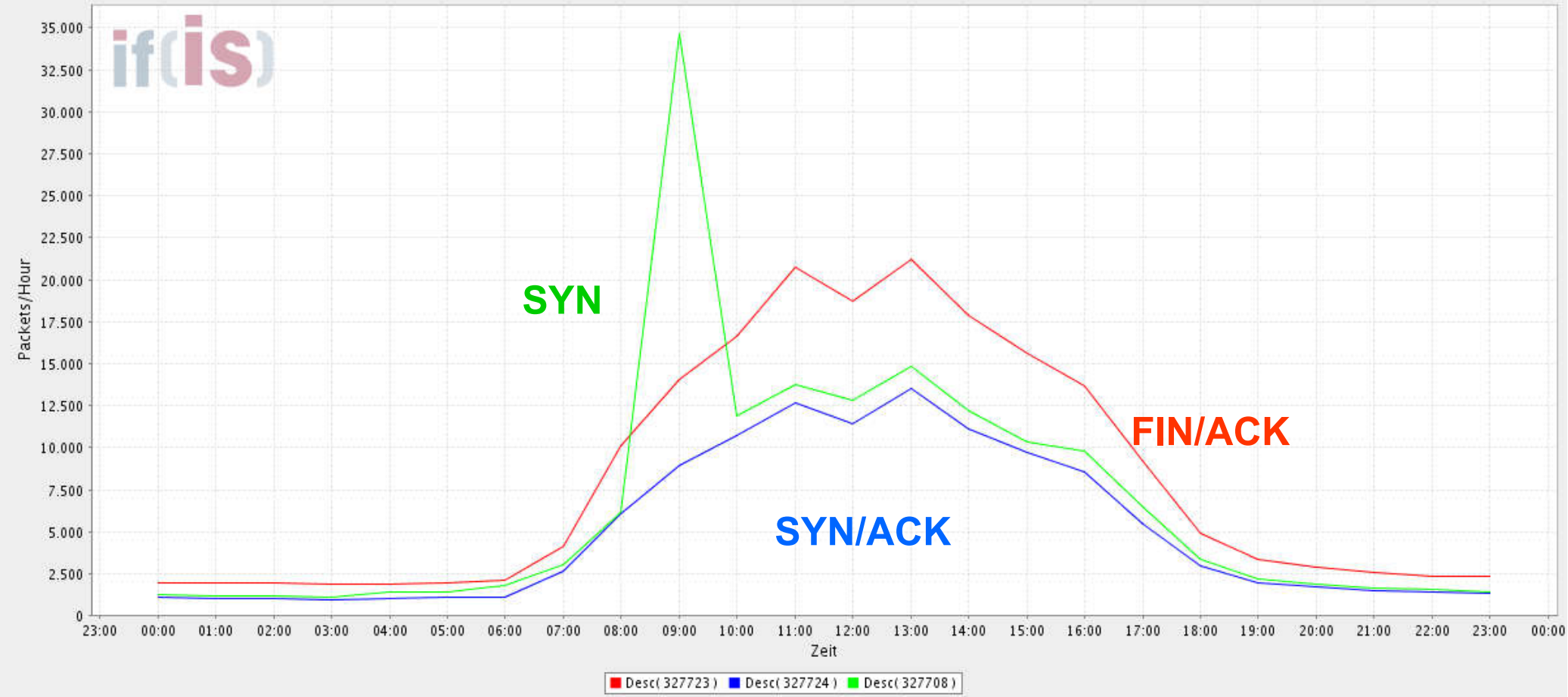
Neuronales Netzwerk

Abweichungen



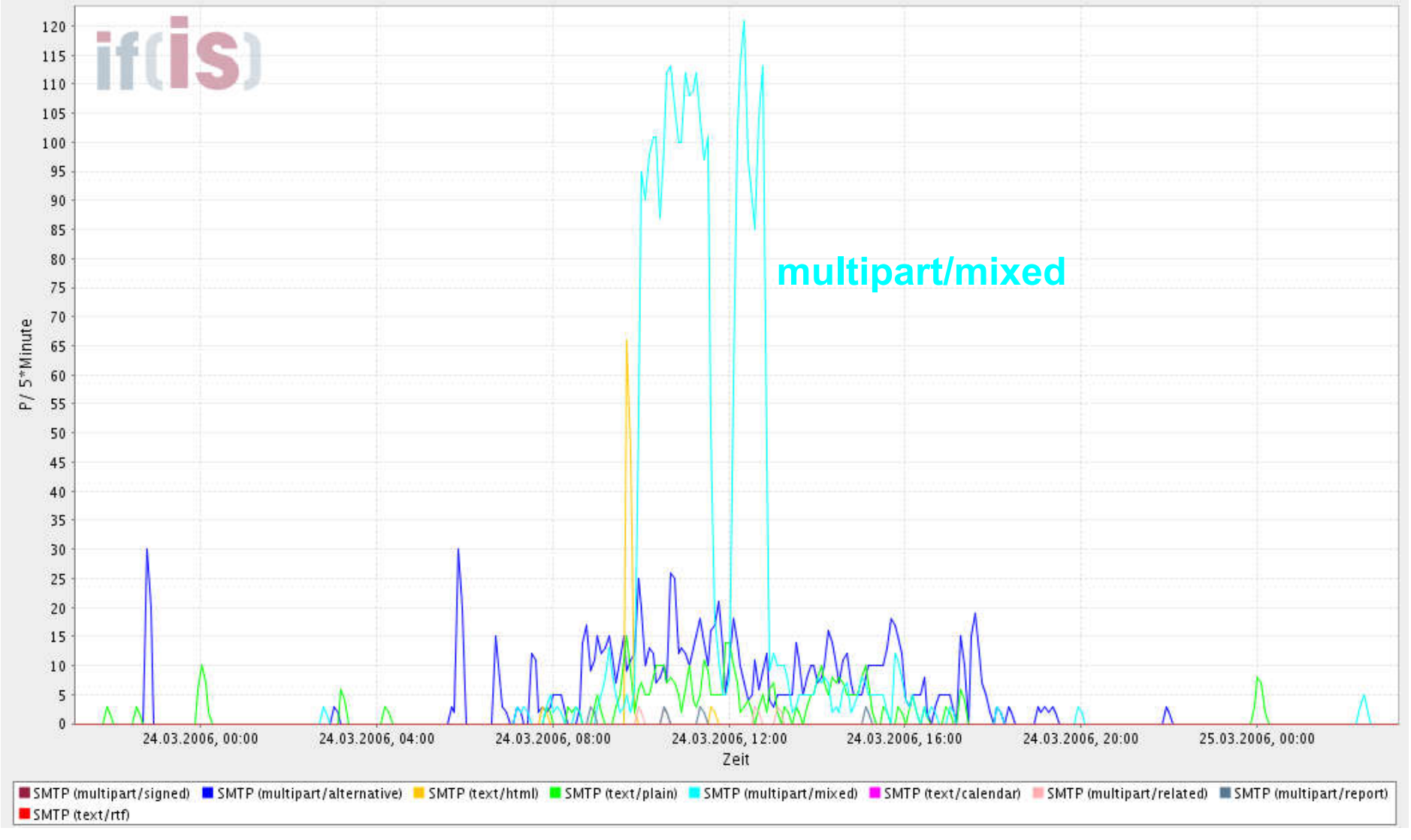
TCP Header „Code Bits“-field
→ Result: Detection of attacks

- **SYN-Scan (Potential Attack)**
  - Period of SYN scan can easily be detected



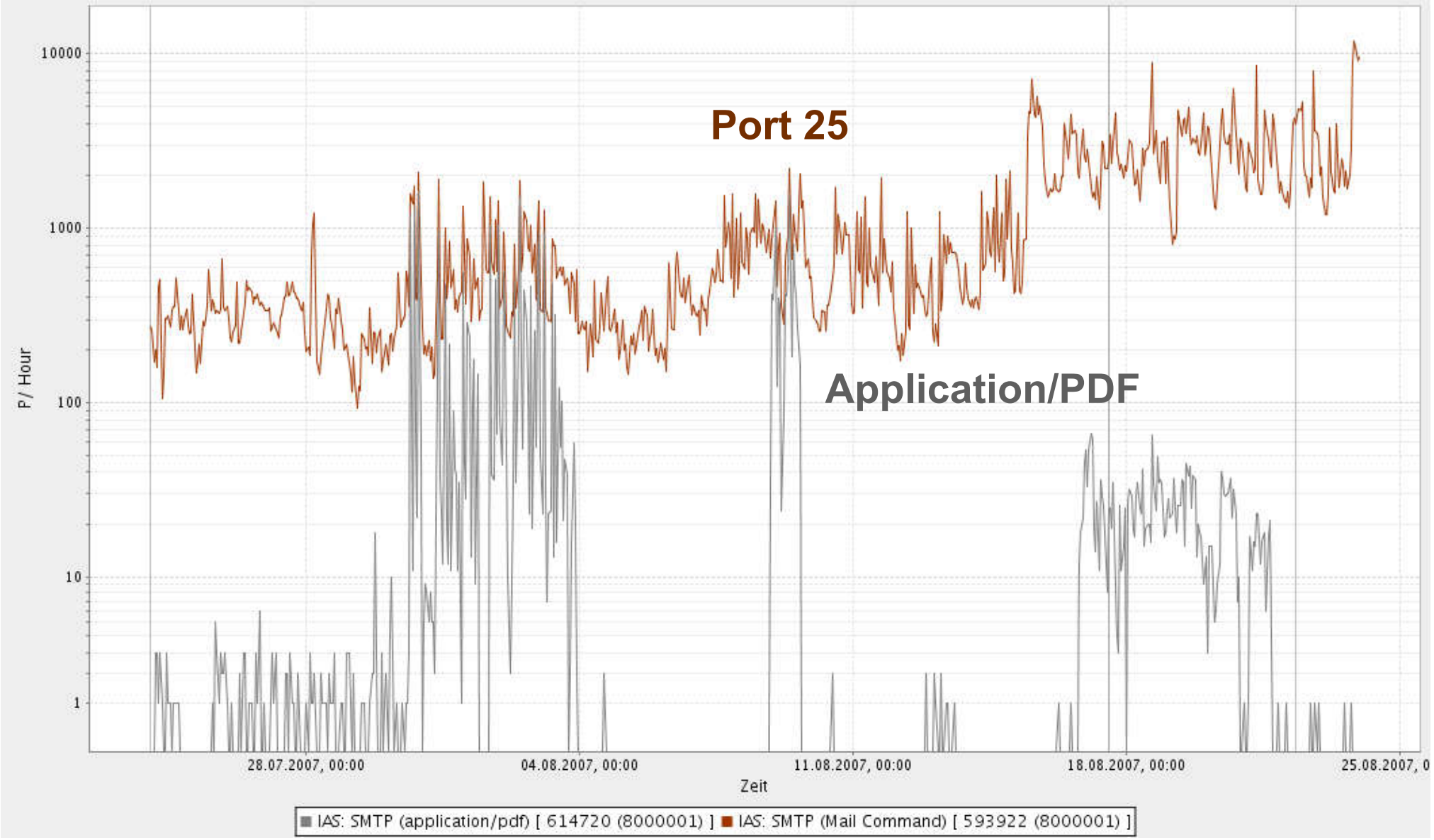
SMTP Header "MIME-Type"
→ Result: Detection of attacks (1/3)

- **SMTP Content Type**
  - Temporarily more e-mails with attachments -> Mail-(Worms/Virus)!



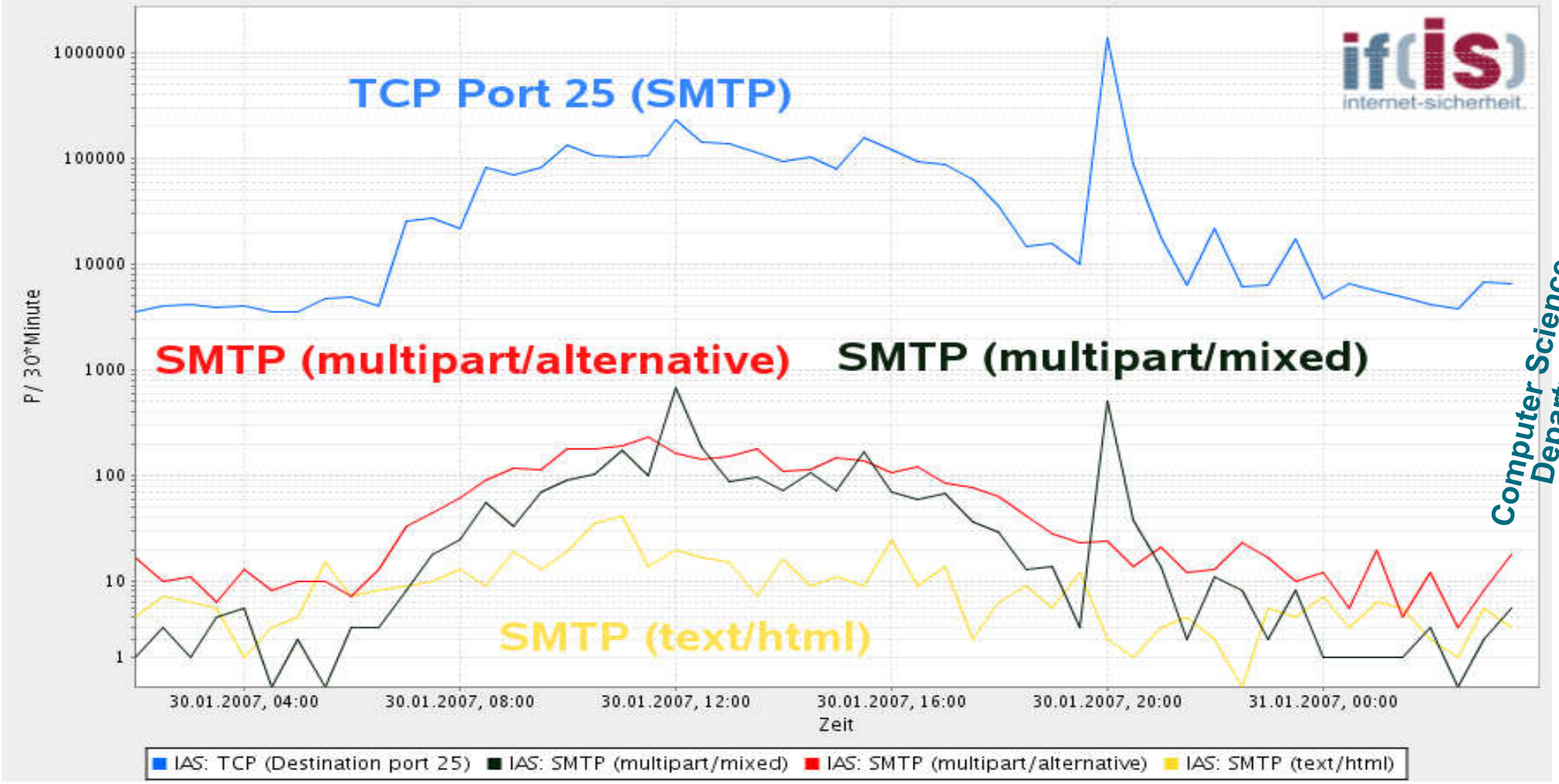
SMTP Header "MIME-Type"
→ Result: Detection of attacks (2/3)

- **PDF Spam Wave**



SMTP Header "MIME-Type"
→ Result: Detection of attacks (3/3)

- **BKA worm (Sober.Z)**
  - The waves were transmitted in January 2007 concentrated at 3 pm and/or 8 pm.

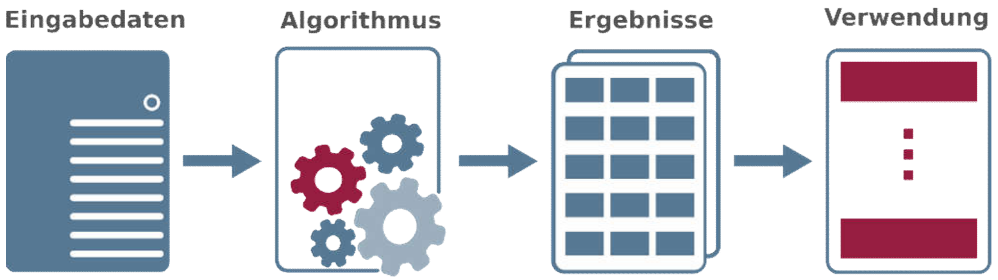


Artificial Intelligence
→ Weitere Beispiele

- Logdatenanalyse
- Malware-Erkennung
- Security Information and Event Management (SIEM)
- Threat Intelligence
- Spracherkennung
- Bilderkennung
- Authentifikationsverfahren
- Identifizierung von Spam-Mails
- IT-Forensik
- …

Artificial Intelligence
→ Probleme in der IT-Sicherheit

- **„Hacker“ greifen an und manipulieren**
  - die Eingabedaten (Input)
  - die Algorithmen
  - die Ergebnisse (Output)
  - die Verwendung



- **„Hacker“ verwenden KI ebenfalls für ihre Zwecke**
  - Schwachstellensuche
  - Passwortknacker
  - Videomanipulation
    - „Fake Obama Video“
    - „Make Putin Smile Video“

Artificial Intelligence
→ Allgemeine Herausforderungen

- Datenschutz
- Selbstbestimmung
- Diskriminierung
- Vertrauenswürdigkeit der Daten und Ergebnisse
- …

Artificial Intelligence
→ Ergebnis und Ausblick

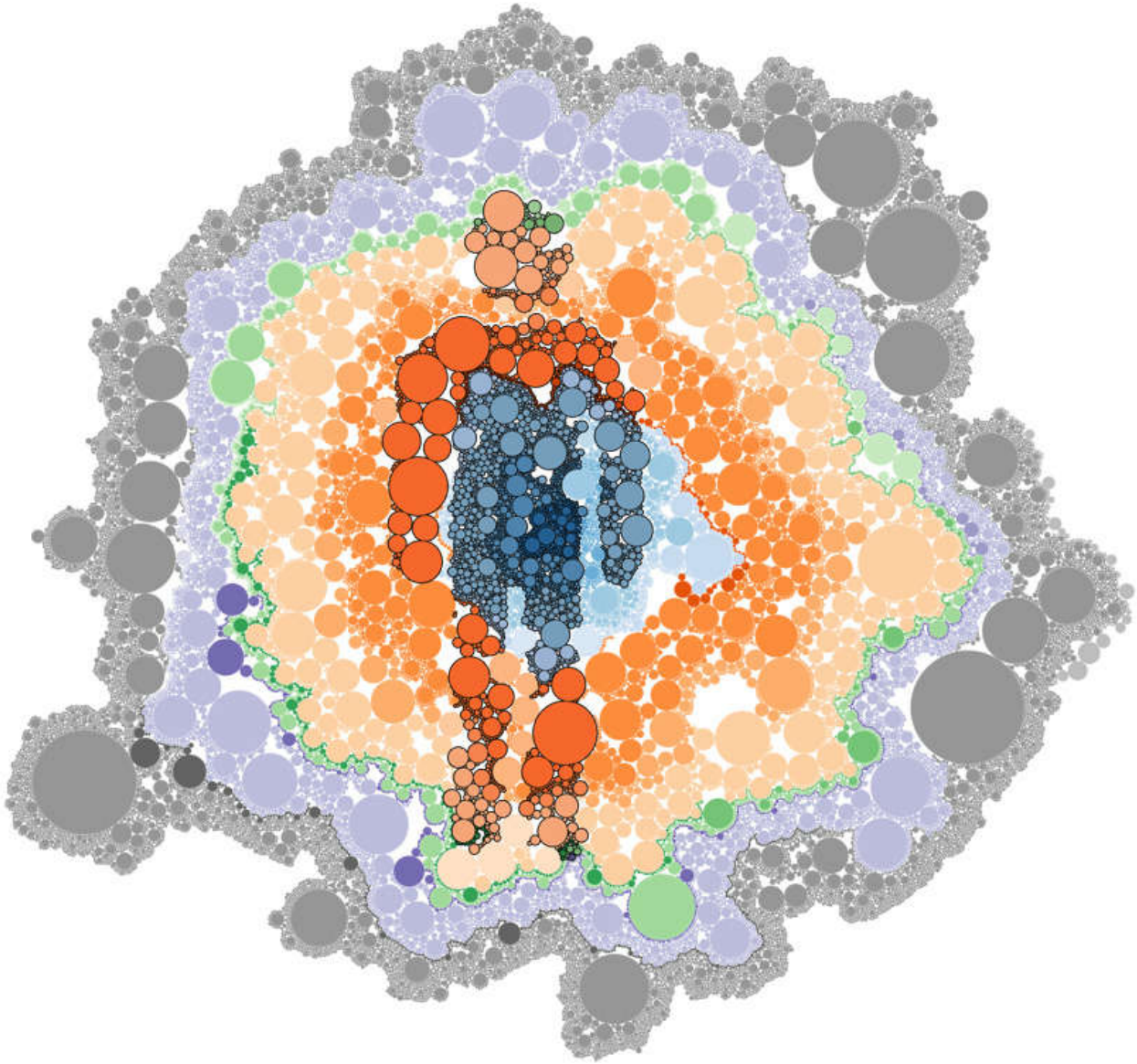
- **Wichtige Technologie für die Zukunft**
  - Erkennen von Angriffen
  - Erkennen von Nutzern
  - …
- **Starke politische Fokussierung**
  - Sehr viel Forschung
  - Sehr viel Förderung
  - …
- **Technologische Souveränität wird immer wichtiger**

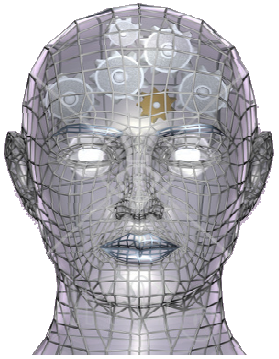
Artificial Intelligence
→ Einstiegspunkte

- **Amazon**
  - Amazon Machine Learning
  - Amazon Lex (Konversationsschnittstellen für Sprache und Text)
- **Microsoft**
  - Azure Machine Learning
  - Microsoft Cognitive Services (Bildanalyse und Gesichtserkennung)
- **Google**
  - Google Cloud Machine Learning Engine
  - Tensorflow
- **IBM**
  - IBM Machine Learning
  - **Watson**

Artificial Intelligence

→ Diskussion „Hype oder Trend“





Westfälische Hochschule
Gelsenkirchen Bocholt Recklinghausen
University of Applied Sciences

Artificial Intelligence

Hype oder Trend?

Mit **Artificial Intelligence** in die Zukunft!

Dr.
Rolf Reinema

Prof. Dr. (TU NN)
Norbert Pohlmann

SIEMENS

if(is)
internet-sicherheit.

Anhang / Credits

Wir empfehlen

- **Kostenlose App securityNews**

- **7. Sinn im Internet (Cyberschutzraum)**
  https://www.youtube.com/channel/UCEMkHjW9dHcWfek_En3xhjg

- **Cybärcast – Der IT-Sicherheit Podcast**
  https://podcast.internet-sicherheit.de/

- **Master Internet-Sicherheit**
  https://it-sicherheit.de/master-studieren/

Besuchen und abonnieren Sie uns :-)

WWW
https://www.internet-sicherheit.de

Facebook
https://www.facebook.com/Internet.Sicherheit.ifis

Twitter
https://twitter.com/_ifis

Google+
https://plus.google.com/107690471983651262369/posts

YouTube
https://www.youtube.com/user/InternetSicherheitDE/

Prof. Norbert Pohlmann
https://norbert-pohlmann.com/

Quellen Bildmaterial

Eingebettete Piktogramme:
- Institut für Internet-Sicherheit – if(is)

Der Marktplatz IT-Sicherheit

(IT-Sicherheits-) Anbieter, Lösungen, Jobs, Veranstaltungen und Hilfestellungen (Ratgeber, IT-Sicherheitstipps, Glossar, u.v.m.) leicht & einfach finden.
https://www.it-sicherheit.de/

Literatur

Artikel:

N. Pohlmann, S. Schmidt: „Der Virtuelle IT-Sicherheitsberater – Künstliche Intelligenz (KI) ergänzt statische Anomalien-Erkennung und signaturbasierte Intrusion Detection“, IT-Sicherheit – Management und Praxis, DATAKONTEXT-Fachverlag, 05/2009

D. Petersen, N. Pohlmann: "Ideales Internet-Frühwarnsystem", DuD Datenschutz und Datensicherheit – Recht und Sicherheit in Informationsverarbeitung und Kommunikation, Vieweg Verlag, 02/2011

M. Fourné, D. Petersen, N. Pohlmann: “Attack-Test and Verification Systems, Steps Towards Verifiable Anomaly Detection“. In Proceedings der INFORMATIK 2013 - Informatik angepasst an Mensch, Organisation und Umwelt, Hrsg.: Matthias Horbach, GI, Bonn 2013

D. Petersen, N. Pohlmann: „Kommunikationslage im Blick - Gefahr erkannt, Gefahr gebannt“, IT-Sicherheit – Management und Praxis, DATAKONTEXT-Fachverlag, 4/2014

U. Coester, N. Pohlmann: „Verlieren wir schleichend die Kontrolle über unser Handeln? Autonomie hat oberste Priorität“, BI-SPEKTRUM Fachzeitschrift für Business Intelligence und Data Warehousing, 05-2015

U. Coester, N. Pohlmann: „Diskriminierung und weniger Selbstbestimmung? Die Schattenseiten der Algorithmen“, tec4u, 12/17

Weitere Artikel siehe: https://norbert-pohlmann.com/artikel/