findic

WHITE PAPER

Anomalien ohne Anleitung

Selbstüberwachte Anomalieerkennung in Transaktions- und Schadendaten

Inhalt

Über dieses White Paper ... 3
1 · Das Label-Problem, das niemand zugibt ... 3
2 · Methoden, von denen wir dachten, sie reichen ... 5
Isolation Forest: treu, aber taub für Kontext ... 5
Gradient Boosting ... 5
Graph-Autoencoder: Das DOMINANT-Erbe ... 5
Sequenzmodelle: Der Transformer, der Buchhaltung gelernt hat ... 6
Kontrastives Pre-training: der Ausweg aus dem Zirkelschluss ... 7
3 · Die Architektur: vier Sichten, eine Entscheidung ... 8
4 · Mit und ohne Antwort: die zwei Betriebsarten ... 9
Was die Aufsicht bringt und was sie kostet ... 9
Was die Selbstüberwachung sieht und was nicht ... 9
Der Graph als Schiedsrichter zwischen beiden Welten ... 10
Die Brücke: Feedback wird zu Aufsicht ... 10
5 · Messen, ohne sich zu belügen ... 12
6 · Bewährungsproben ... 14
7 · Wo sind die Grenzen? ... 14
8 · Was ein Pilot bei uns bedeutet ... 15
9 · Quellen / weiterführende Literatur ... 16
10 · Kontakt ... 18

Über dieses White Paper

Die Erkennung von Betrug und Geldwäsche auf Transaktions- und Schadensdaten wirkt auf den ersten Blick wie ein klassisches Modellierungsproblem: auffällige Fälle identifizieren und Risiken bewerten. In der Praxis ist die Aufgabe deutlich schwieriger. Bestätigte Betrugsfälle sind selten, vorhandene Labels oft verzerrt, Regelwerke altern schnell, und viele auffällige Muster entstehen erst im Zusammenspiel von Konten, Verträgen, Zahlungsflüssen und zeitlichen Abläufen. Dort stoßen einfache Schwellenwerte und isolierte Einzelmodelle an ihre Grenzen.

Dieses White Paper beschreibt, wie unsere Anomalieerkennung solche Risiken aus mehreren Perspektiven analysiert, nämlich über transaktionsnahe Merkmale, Graphstrukturen, Sequenzen und nachvollziehbare Regeln. Dabei steht im Mittelpunkt nicht das Versprechen einer automatischen Betrugsentscheidung, sondern eine belastbare Priorisierung von Prüffällen. Es soll zeigen, welche Modellierungsentscheidungen dafür notwendig sind, warum authentische Scores wichtiger sind als scheinbar präzise Wahrscheinlichkeiten und wo die Grenzen solcher Systeme im Betrieb liegen.

1 · Das Label-Problem, das niemand zugibt

Betrugserkennung wird in der Literatur gern als Klassifikationsproblem behandelt. Man nehme gelabelte Transaktionen, trainiere ein Modell, messe AUC. In der Praxis sieht die Ausgangslage fast immer anders aus. Das ist kein Randphänomen, sondern das bekannte Grundproblem überwachter Verfahren:[1] verlässliche Labels existieren schlicht nicht. Was stattdessen vorliegt, ist veraltet, verzerrt und geprägt von einem Meldewesen, in dem „Betrug" das bedeutet, was ein Sachbearbeiter zu einem bestimmten Zeitpunkt vermutete. Bestätigte Fälle sind selten, dabei ist ein Prozent schon viel. Man findet nur die Muster, nach denen man gesucht hat.

Projekte umschiffen das auf unterschiedliche Arten. Drei davon sind Sackgassen.

Der Regelweg. Regelbasierte Systeme erwecken den Eindruck von Nachvollziehbarkeit und Revisionssicherheit. Jedoch sind sie blind für alles, was bei der Formulierung der Regel noch nicht bekannt war. Regelwerke altern schlecht. Sie beantworten die Fragen von gestern mit wachsender Präzision.

Der Pseudo-Label-Weg. Man nimmt ein unsupervisiertes Modell, erklärt die obersten fünf Prozent seiner Scores zu „Betrug" und trainiert ein zweites, supervisiertes Modell darauf. Das liest sich zunächst wie ein pragmatischer Umweg, erzeugt aber einen Zirkelschluss, sobald die Pseudo-Labels ungeprüft als Bodenwahrheit behandelt werden. Das zweite Modell lernt, die Meinung des ersten zu reproduzieren. Mit menschlicher Gegenprüfung, Confidence-Filterung oder iterativer Nachlabelung lässt sich dieser Ansatz retten. Ohne solche Korrekturen wird jede systematische Blindheit des ersten Modells nicht korrigiert, sondern verstärkt. Hinterher sieht die Evaluation gut aus, weil Modell zwei ja „vorhersagt", was Modell eins bereits vorgegeben hat. Wir haben diese Bauart in zwei fremden Codebases vorgefunden, inklusive einer Kalibrierung auf den Pseudo-Labels, die den Scores den Anschein echter

[1] Hilal, W., Gadsden, S. A. & Yawney, J. (2022), „Financial Fraud: A Review of Anomaly Detection Techniques and Recent Advances", Expert Systems with Applications 193

Wahrscheinlichkeiten gab. So kann in der Praxis eine Sicherheit entstehen, die fachlich nicht belastbar ist.

Der synthetische Weg. Statt Labels zu erfinden, erfindet man die Fälle gleich selbst. Interpolationsverfahren wie SMOTE erzeugen künstliche Minderheitenfälle entlang der Verbindungslinien bekannter Betrugsfälle,[2] generative Modelle wie CTGAN lernen die Verteilung und sampeln neue Transaktionen.[3] Das adressiert die Seltenheit, aber nicht die Verzerrung: Wer aus hundert bekannten Fällen tausend neue rechnet, multipliziert die Ermittlungsgeschichte, statt sie zu erweitern. Interpolation kennt zudem keine Fachlogik. Sie mischt Betrugsarten zu Fällen, die nie existierten, und die erzeugten Punkte fallen nachweisbar häufig in die Regionen der Mehrheitsklasse, in die sie nicht gehören.[4] Bei generativen Modellen verschärft sich das Bild. Die jüngsten Benchmarks zeigen, dass synthetische Transaktionsdaten statistisch täuschend echt aussehen können, während genau die Verhaltensstrukturen zerstört werden, auf die es bei der Betrugserkennung ankommt.[5] Ein Modell, das auf solchen Daten glänzt, hat die Annahmen des Generators gelernt, nicht die Muster der Täter.

Unser Weg ist unbequemer, aber lohnt sich: selbstüberwachte Verfahren, die aus der Struktur der Daten selbst lernen, was normal ist, und damit umgehen, dass ihre Ausgaben Ränge sind, keine Wahrscheinlichkeiten. Hier setzt findic an. Wer keine Labels hat, darf keine Wahrscheinlichkeiten vortäuschen, und wer welche hat, muss sie für Kalibrierung und Evaluation reservieren statt für das Training.

Zuerst aber zu den Begrifflichkeiten, bevor es technisch wird. Unüberwacht nennen wir jedes Verfahren, das ohne externe Zielvariable auskommt. Selbstüberwacht nennen wir die Teilmenge davon, die ihr Trainingssignal aus den Daten selbst konstruiert. Der maskierte Transformer und der kontrastive Encoder sind selbstüberwacht, der Isolation Forest ist unüberwacht, und die Regelsicht ist kein Lernverfahren. Bewertet wird immer eine Einheit, je nach Projekt die Transaktion oder der Schadenfall. Knoten im Graphen sind Konten beziehungsweise Vertragspartner, Kanten sind Zahlungsflüsse oder begründete Ähnlichkeitsbeziehungen. Wenn wir in diesem Paper „Fall" sagen, meinen wir die bewertete Einheit des jeweiligen Projekts.

[2] Chawla, N. V., Bowyer, K. W., Hall, L. O. & Kegelmeyer, W. P. (2002), „SMOTE: Synthetic Minority Over-sampling Technique", Journal of Artificial Intelligence Research 16, S. 321–357

[3] Xu, L., Skoularidou, M., Cuesta-Infante, A. & Veeramachaneni, K. (2019), „Modeling Tabular Data using Conditional GAN", NeurIPS 2019

[4] Tarawneh, A. S. et al. (2022), „Stop Oversampling for Class Imbalance Learning: A Critical Review", IEEE Access

[5] Sajja, B. (2026), „Synthetic Tabular Generators Fail to Preserve Behavioral Fraud Patterns: A Benchmark on Temporal, Velocity, and Multi-Account Signals", arXiv:2604.13125

2 · Methoden, von denen wir dachten, sie reichen

Bevor wir zur Architektur kommen, ein paar Erfahrungen aus dem Maschinenraum. Sie erklären, warum das System so aussieht, wie es aussieht.

Isolation Forest: treu, aber taub für Kontext

Der Isolation Forest[6] ist das Arbeitstier jeder Anomaliepipeline, und er bleibt bei uns im Ensemble. Das ist so aus gutem Grund gewählt. Seine Grenze ist konzeptionell. Er sieht jede Transaktion isoliert. Ein Betrag von 9.800 € ist für ihn entweder auffällig oder nicht. Dass derselbe Kunde in der Vorwoche schon zweimal 9.700 € und 9.850 € bewegt hat, entgeht ihm, weil ihm jede Form von Gedächtnis fehlt. Structuring, das klassische Geldwäschemuster unter der Meldeschwelle, nutzt geradezu diese Blindheit, präziser: die Blindheit des Merkmalsraums. Wer Velocity-Zähler, kumulierte Beträge und Schwellenabstände explizit als Merkmale kodiert, macht Structuring auch für den Isolation Forest sichtbar. Nur formuliert hat diese Merkmale dann ebenfalls ein Mensch.

Gradient Boosting

Es ist inzwischen gut dokumentiert, dass baumbasierte Verfahren auf typischen Tabellendaten Deep-Learning-Architekturen schlagen, systematisch, über Dutzende Datensätze und großzügige Hyperparameter-Budgets hinweg.[7] Unsere Erfahrung deckt sich damit, mit einer Präzisierung. Auf den transaktionsnahen Merkmalen (Beträge, Zeitabstände, Zähler, Verhältnisse) ist LightGBM bei uns das stärkste Einzelmodell,[8] sobald irgendwelche Labels existieren. Für solche Daten ist ein tiefes Netz als Hauptmodell daher nur schwer zu rechtfertigen, wenn der zusätzliche Kontext nicht ausdrücklich genutzt wird.

Aber Boosting kennt weder Nachbarschaft noch Reihenfolge. Es gibt keine Baumstruktur, die „dieser Begünstigte hängt an einem Ring aus 14 nahezu identischen Konten" ausdrückt, ohne dass jemand dieses Muster vorher als Merkmal formuliert hat. Dort setzen Graph- und Sequenzmodelle an.

Graph-Autoencoder: Das DOMINANT-Erbe

Die Idee, Anomalien über Rekonstruktionsfehler eines Graph-Autoencoders zu finden, ist seit DOMINANT (SDM 2019) etabliert:[9] Ein Graph-Convolution-Encoder bettet jeden Knoten ein, zwei Decoder rekonstruieren Attribute bzw. Adjazenz. Dort, wo beides schlecht gelingt, liegt die Anomalie. Varianten wie AnomalyDAE zerlegen Attribut- und Strukturraum noch sauberer.[10] Wir haben dieses Grundmuster übernommen und erweitert: Attention statt Convolution (Graph Attention Networks erlauben es, die Nachbarschaft je nach Relevanz zu gewichten[11]), mehrere Relationen parallel

---

[6] Liu, F. T., Ting, K. M. & Zhou, Z.-H. (2008), „Isolation Forest", Proceedings of the 8th IEEE International Conference on Data Mining (ICDM 2008), S. 413–422

[7] Grinsztajn, L., Oyallon, E. & Varoquaux, G. (2022), „Why do tree-based models still outperform deep learning on typical tabular data?", NeurIPS 2022

[8] Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., Ye, Q. & Liu, T.-Y. (2017), „LightGBM: A Highly Efficient Gradient Boosting Decision Tree", Advances in Neural Information Processing Systems 30

[9] Ding, K., Li, J., Bhanushali, R. & Liu, H. (2019), „Deep Anomaly Detection on Attributed Networks", Proceedings of the 2019 SIAM International Conference on Data Mining (SDM 2019)

[10] Fan, H., Zhang, F. & Li, Z. (2020), „AnomalyDAE: Dual Autoencoder for Anomaly Detection on Attributed Networks", Proceedings of ICASSP 2020

[11] Veličković, P., Cucurull, G., Casanova, A., Romero, A., Liò, P. & Bengio, Y. (2018), „Graph Attention Networks", ICLR 2018

(Merkmalsähnlichkeit, Gemeinschaftsnähe, echte Zahlungsflüsse) mit lernbarer Gewichtung, und, das ist der Teil, der in der Literatur kaum beschrieben ist, eine Fehlerkombination, die mit Median und MAD statt mit Mittelwert und Standardabweichung normalisiert. Rekonstruktionsfehler sind typischerweise heavy-tailed; die robuste Skalierung verhindert, dass wenige extreme Werte die gesamte Score-Skala dominieren.

Was wir dabei gelernt haben: Der Graph ist selbst eine Modellannahme. In einer frühen Version verband unser k-nächste-Nachbarn-Graph aufgrund eines Sortierfehlers, der monatelang unentdeckt blieb, die Knoten, mit ihren unähnlichsten statt ihren ähnlichsten Nachbarn. Das System lieferte trotzdem plausible Scores, weil Autoencoder erstaunlich fehlertolerant trainieren. Erst bei einem Kontrollgang durch die Kantenliste fiel uns der Fehler auf. Seitdem gilt bei uns, jede Graphkonstruktion ist Code, der genauso reviewed wird wie das Modell. Und jede Relation, die wir aufspannen, muss inhaltlich begründet sein. Unsere „Mule-Relation" verbindet Konten, die sich auf den hochvarianzesten Merkmalen auffällig ähneln, weil das das Muster ist, mit dem Strohmann-Netzwerke in der Geldwäsche arbeiten.[12] Konkret: Wir bestimmen die Merkmale mit der höchsten Varianz im jeweiligen Bestand, standardisieren sie robust und verbinden Konten über einen k-nächste-Nachbarn-Graphen auf diesen Merkmalen. Merkmalszahl und Nachbarschaftsgröße sind dokumentierte Projektparameter. Die vollständige Spezifikation der Kantengewichte veröffentlichen wir nicht. Wichtig ist uns hier das Prinzip. Jede Relation ist Code mit einer inhaltlichen Begründung, keine Konfiguration, die „irgendwie sinnvoll" wirkte.

Ein zweites Detail aus dem Maschinenraum gehört bewusst hierher und nicht zum Isolation Forest: die Skalierung. Wir nutzten robuste Skalierer (Median, IQR), bis ein Datensatz mit Beträgen im Faktor-100-Bereich zeigte, dass „robust" nicht „gebunden" heißt. Die Extremwerte dominierten nach der Transformation weiterhin jede Distanz und damit jeden k-nächste-Nachbarn-Graphen. In unserem Datensatz erwies sich der Umstieg auf eine Quantiltransformation auf Normalverteilung als günstiger, weil sie alle Merkmale in vergleichbare Bereiche band. Die Transformation komprimiert gerade die Ränder der Verteilung, und in diesen Rändern steckt bei Betrugsdaten oft das Signal. Grinsztajn und Kollegen setzen die Quantiltransformation auf Normalverteilung in ihrem großen Tabellar-Benchmark als Vorverarbeitung ein.[13] Unsere Beobachtung geht in dieselbe Richtung, wobei unsere Erfahrung zeigt, dass sie auch für die distanzbasierte Graphkonstruktion zählt. Für den Isolation Forest ist die Frage dagegen weitgehend irrelevant. Seine Splits beruhen auf Reihenfolgen, und monotone Transformationen ändern an Reihenfolgen nichts.

Sequenzmodelle: Der Transformer, der Buchhaltung gelernt hat

Die jüngste Erweiterung ist ein Transformer über der Transaktionshistorie jedes Kontos. Das Training folgt dem Masking-Prinzip, das BERT für Sprache etabliert hat.[14] Ein zufälliger Teil der Merkmale einer zufälligen Teilmenge der Transaktionen wird ausgeblendet, das Modell rekonstruiert ihn aus dem Kontext davor und danach. Wer das Muster eines Kontos verinnerlicht hat, kann gut rekonstruieren. Ein Konto, das plötzlich aus seinem Verhaltensmuster fällt, erzeugt hohe Fehler. Damit liefert die Sequenzsicht das Signal, das dem Isolation Forest fehlt: ein Gedächtnis. Eine Präzisierung, die im Betrieb

[12] Dou, Y., Liu, Z., Sun, L., Deng, Y., Peng, H. & Yu, P. S. (2020), „Enhancing Graph Neural Network-based Fraud Detectors against Camouflaged Fraudsters", Proceedings of CIKM 2020

[13] Grinsztajn, L., Oyallon, E. & Varoquaux, G. (2022), „Why do tree-based models still outperform deep learning on typical tabular data?", NeurIPS 2022

[14] Devlin, J., Chang, M.-W., Lee, K. & Toutanova, K. (2019), „BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding", Proceedings of NAACL-HLT 2019

zählt. Die bidirektionale Rekonstruktion nutzt Kontext davor und danach. Das ist zulässig, solange rückblickend geprüft wird, also alle Transaktionen eines abgeschlossenen Zeitraums gleichzeitig bewertet werden. Für ein Scoring in Echtzeit darf das Modell keine Zukunft sehen; in dieser Betriebsart maskieren wir kausal und verwenden nur den Kontext davor. Beide Modi sind implementiert, und wir deklarieren je nach Einsatzszenario, welcher läuft.

Aus dem Betrieb kamen dazu zwei wichtige Hinweise. Erstens: Auf unskalierten Rohwerten stieg der Rekonstruktionsverlust beim ersten Lauf auf Werte jenseits von 10^{19}, weil der quadratische Fehler auf Euro-Beträgen jede Normalisierung verspottet. Zweitens: Wir decken die letzten 32 Ereignisse pro Konto ab. Das stellt einen Kompromiss zwischen Gedächtnistiefe und Padding-Anteil bei Konten mit wenig Bewegung dar, nicht das Ergebnis einer systematischen Hyperparameter-Suche. Schnellbetrug über Wochen hinweg entzieht sich dieser Fensterlogik und bleibt Sache des Graphen.

Kontrastives Pre-training: der Ausweg aus dem Zirkelschluss

Erinnern wir uns an die Pseudo-Label-Falle aus Abschnitt 1. Die moderne Antwort darauf heißt kontrastives Lernen: Statt sich Labels auszudenken, lernt der Encoder, dass zwei leicht gestörte Varianten derselben Transaktion zusammengehören (Feature-Dropout hier, Gaußsches Rauschen da) und sich von allen anderen Transaktionen unterscheiden. Auf Graphen hat diese Familie mit Verfahren wie SL-GAD gezeigt, dass sie Anomalien ohne ein einziges Label sichtbar macht, indem sie generative und kontrastive Signale kombiniert.[15] Wir pre-trainieren den GAT-Encoder mit einer NT-Xent-Verlustfunktion,[16] bevor der eigentliche Autoencoder trainiert wird. Der Effekt im Vergleich zur alten Pseudo-Label-Pipeline ist qualitativ. Die Repräsentationen sind glatter, die Scores stabiler gegenüber der Wahl des Graphen, und niemand musste sich Fälle erfinden.

[15] Zheng, Y., Jin, M., Liu, Y., Chi, L., Phan, K. T. & Chen, Y.-P. P. (2021), „Generative and Contrastive Self-Supervised Learning for Graph Anomaly Detection", IEEE Transactions on Knowledge and Data Engineering

[16] Chen, T., Kornblith, S., Norouzi, M. & Hinton, G. (2020), „A Simple Framework for Contrastive Learning of Visual Representations", *Proceedings of ICML 2020*

3 · Die Architektur: vier Sichten, eine Entscheidung

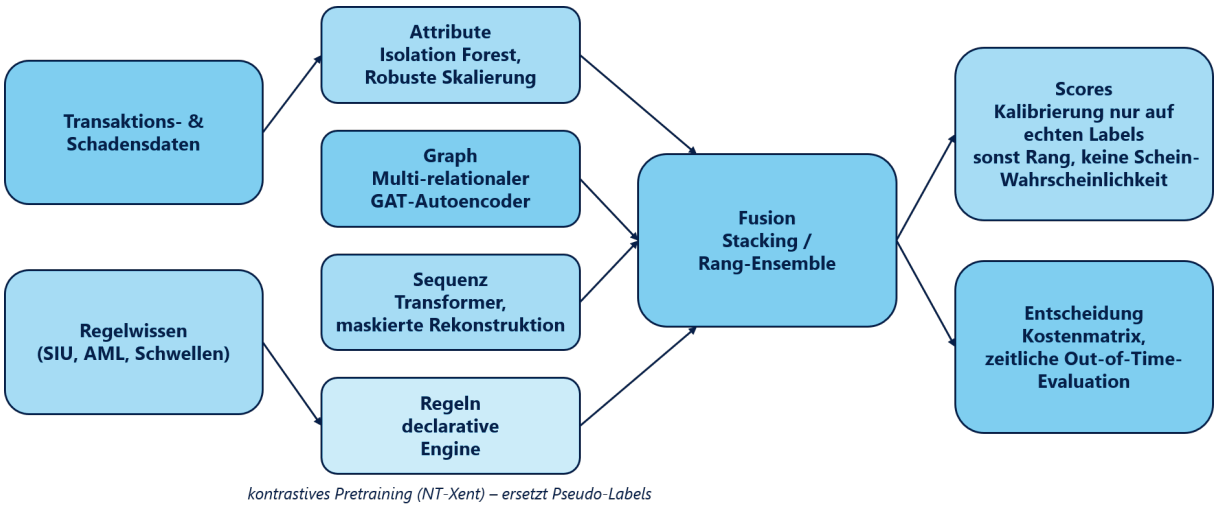

Abbildung 1: Die vier Sichten von findic Anomalieerkennung auf eine Transaktion. Keine Einzelmethode ist für die Gesamtaussage verantwortlich; die Fusion entscheidet.

Vier Sichten treffen aufeinander, und keine darf allein entscheiden: die Attributsicht (Isolation Forest auf robust skalierten Merkmalen), die Graphsicht (der multi-relationale GAT-Autoencoder), die Sequenzsicht (der maskierte Transformer) und die Regelsicht, bewusst nicht abgeschafft, weil Regeln das einzige Modul sind, das fachliches Vorwissen direkt und auditierbar aufnimmt. Die Fusion erfolgt als Stacking. Existieren echte Labels, lernt ein LightGBM die Kombination. Existieren keine, wird deterministisch über Ränge gemittelt – sieben Rangfolgen, keine davon zufällig und kein Meta-Learner, der sich selbst belügt.

Zwei Entscheidungen in diesem Bild verdienen einen zweiten Blick, weil sie gegen die Konvention gehen.

Wir normalisieren über Ränge, nicht über Z-Scores pro Batch. Der Grund liegt in der Fusion: Ränge machen die unterschiedlich skalierten Scores der Sichten überhaupt erst kombinierbar, und sie sind robust gegenüber monotoner Verzerrung einzelner Score-Skalen. Eine Einschränkung benennen wir offen. Ränge sind relativ zur jeweiligen Population. Ändert sich der Datenbestand, kann der Rang eines Kunden wandern, ohne dass sich sein Verhalten geändert hat. Das ist dieselbe Kritik, die an pro Batch neu geschätzten Z-Scores berechtigt ist. Wer zeitlich stabile Schwellen braucht, kommt um eine feste Referenzverteilung und Kalibrierung auf bestätigtem Feedback nicht herum. Deshalb endet unsere Rampe dort.

Wir verzichten auf eingestreutes Zufallsrauschen. In der Codebase, aus der die Anomalieerkennung von findic hervorging, fand sich ein Fallback, der bei entarteten Score-Verteilungen deterministisch durch Zufallszahlen ersetzte. Das ist praktisch, damit die Pipeline nie abstürzt. Es ist zugleich problematisch, weil Scores nicht mehr reproduzierbar waren. Reproduzierbarkeit bei gleichem Input ist eine zentrale Voraussetzung für eine revisionsfähige Pipeline – neben der Versionierung von Modell, Daten, Features, Graph, Parametern und Schwellen. Das bedeutet an manchen Stellen, einen Fehler sichtbar werden zu lassen, statt ihn durch eine scheinbar gültige Score-Verteilung zu verdecken.

4 · Mit und ohne Antwort: die zwei Betriebsarten

Die erste Frage in jedem Erstgespräch lautet: „Haben Sie Labels?" Die bessere Frage wäre: „Was würden Sie mit ihnen anfangen?" Denn supervised und unsupervised sind bei Anomalieerkennung keine Lager, zwischen denen man sich entscheidet, sondern zwei Betriebsarten ein- und derselben Pipeline mit verschiedenen Stärken, verschiedenen Versagensmustern und einer Brücke dazwischen. Dieses Kapitel legt dar, was sich wirklich ändert, wenn die Antworten auf die Daten treffen.

Was die Aufsicht bringt und was sie kostet

Supervisiertes Lernen bringt drei Vorteile, die unsupervisiertes nicht liefern kann. Erstens **Wahrscheinlichkeiten**: Nur wer weiß, was tatsächlich Betrug war, kann Scores an beobachteten Häufigkeiten messen und kalibrieren. Zweitens **Messbarkeit**: PR-AUC, Precision@k, kostenoptimierte Schwellen. Alle Kennzahlen aus dem nächsten Kapitel setzen voraus, dass es eine Wahrheit gibt, gegen die man rechnet. Drittens **Erklärbarkeit**: Mit Labels trainiert unser Stacking-LightGBM auf den sieben Basis-Scores, und SHAP[17] sagt uns, welche dieser Eingangsscores die Entscheidung des Meta-Modells trug. Das ist eine Erklärung der Modellentscheidung, keine kausale Erklärung des Falls. Es ist der Unterschied zwischen „das Modell sagt" und „hier steht, warum".

Der Preis dafür wird selten diskutiert. Labels in Betrugsdaten sind keine Bodenwahrheit, sie sind eine **Befundgeschichte**. Sie dokumentieren, wonach Ermittler gesucht und was sie gefunden haben, nicht, was alles existierte. Ein SIU-Labelbestand ist die Landkarte der bisherigen Ermittlungsstrategie. Ein Modell, das darauf trainiert, lernt die Vergangenheit der Ermittlungen, nicht die Gegenwart. Dazu kommt die Alterung. Wer 2023 bestätigt wurde, beschreibt Muster von 2022. Auf solchen Daten glänzt supervisiertes Lernen bei der Wiedererkennung bekannter Muster. Es hat Nachteile dort, wo Täter ihr Verhalten geändert haben.

Was die Selbstüberwachung sieht und was nicht

Der unsupervisierte Modus kehrt die Bilanz um. Rekonstruktionsfehler, Isolationsscores und kontrastive Repräsentationen kennen keine Ermittlungsgeschichte und können deshalb Muster anzeigen, nach denen nie jemand gesucht hat. Der Strohmann-Ring aus Abschnitt 2 war so ein Fall. Die Blindheit liegt woanders: Ohne Labels gibt es keine Wahrscheinlichkeit, keine Recall-Zahl und vor allem **keine schnelle Antwort auf die Frage „funktioniert es?"**. Die Antwort entsteht erst im Betrieb, Fall für Fall, durch die Prüfung der Top-Ränge.

Und der unsupervisierte Modus hat ein eigenes, wohlbekanntes Versagensmuster: die Flut legitimer Ungewöhnlichkeiten. Großeigentümer mit dreißig Verträgen, Saisonspitzen im Jahresgeschäft, Sanierungsfälle nach Stürmen. In bisherigen Einsätzen bestand die obere Risikoliste im ersten Monat etwa zur Hälfte aus solchen Auffälligkeiten. Der Umgang damit ist keine Modellfrage, sondern eine Kommunikations- und Feedbackfrage. Jeder wegargumentierte Fehlalarm muss ins System zurückfließen, sonst wird er nächste Woche erneut gemeldet.

[17] Lundberg, S. M. & Lee, S.-I. (2017), „A Unified Approach to Interpreting Model Predictions", *Advances in Neural Information Processing Systems 30 (NeurIPS 2017)*

Der Graph als Schiedsrichter zwischen beiden Welten

Ausgerechnet die Graphsicht zeigt den Unterschied der Betriebsarten am schärfsten. Im unsupervisierten Modus ist der GAT-Autoencoder **Richter und Zeuge zugleich**. Er definiert über die Rekonstruktion, was normal ist, und verurteilt Abweichungen. Das ist mächtig und riskant, denn wer den Graphen falsch baut, baut die Normalität falsch, und niemand merkt es, weil es kein Label gibt, das widerspricht. Im supervisierten Modus degradiert sich derselbe Graph-Encoder zum **Zeugen**. Seine Scores werden zu einem von sieben Merkmalen im Stacking, und das Meta-Modell entscheidet anhand echter Fälle, wie viel sein Zeugnis wert ist. Beobachtung aus der Praxis: Der Graph-Score gehört fast immer zu den zwei gewichtigsten SHAP-Treibern.

Bewusst trainieren wir den Graphen selbst **nicht** supervisiert, obwohl das technisch möglich wäre. Für direkte Klassifikation auf Graphen sind bestätigte Fallzahlen, selten mehr als einige hundert, zu wenig Substanz für ein Netz mit Tausenden von Parametern. Das Ergebnis wäre auswendig gelernte Ermittlungsgeschichte. Die Arbeitsteilung hat sich bewährt. Der Graph lernt selbstüberwacht die Geographie der Normalität, das kleine, schnelle Boosting-Modell darüber lernt supervisiert die Entscheidung. Wer auf diesem Gebiet mehr aus den wenigen Positiven holen will, landet übrigens methodisch bei PU-Learning, Lernen aus Positiven und Ungelabelten, einer Literatur, die wir aufmerksam verfolgen, aber für den Produktiveinsatz noch nicht für reif halten.[18]

Die Brücke: Feedback wird zu Aufsicht

Zwischen den beiden Betriebsarten gibt es keinen Sprung, sondern eine Rampe. Sie ist Teil des Produkts. Ermittler markieren im Alltag bestätigte Treffer und Fehlalarme. Die Rückmeldungen werden über stabile Transaktions-IDs gespeichert, nicht über Zeilenpositionen. Das haben wir gelernt, nachdem ein Re-Import einmal sämtliche Markierungen auf die falschen Fälle geschoben hatte. Die Kalibrierung schaltet von Rang- auf Wahrscheinlichkeitsausgabe um, sobald genügend bestätigte Fälle vorliegen, um eine Kalibrierungskurve mit vertretbarer Unsicherheit zu schätzen. Die Mindestmenge legen wir projektbezogen fest, anhand von Fallzahl, Basisrate und geforderter Präzision, nicht nach einer pauschalen Regel. Das supervisierte Stacking folgt, sobald die Out-of-Time-Validierung zeigt, dass das Meta-Modell die deterministische Rangmittelung tatsächlich schlägt. Das, was sich ändert, ist allein die Entscheidungsschicht. Für Kunden heißt das, der Einstieg ohne Labels ist keine Sackgasse mit Ablaufdatum, sondern die erste Etage desselben Gebäudes. Wir unterscheiden vier Phasen ein und derselben Pipeline – *label-free discovery* (keine Rückmeldungen, reine Rangfolge), *feedback-driven ranking* (Investigator-Markierungen fließen zurück), *calibrated ranking* (Wahrscheinlichkeiten auf bestätigtem Feedback) und *supervised fusion* (Meta-Modell auf echten Labels). Die Architektur bleibt über alle Phasen identisch; was reift, ist allein die Entscheidungsschicht.

[18] Elkan, C. & Noto, K. (2008), „Learning Classifiers from Only Positive and Unlabeled Data", Proceedings of KDD 2008

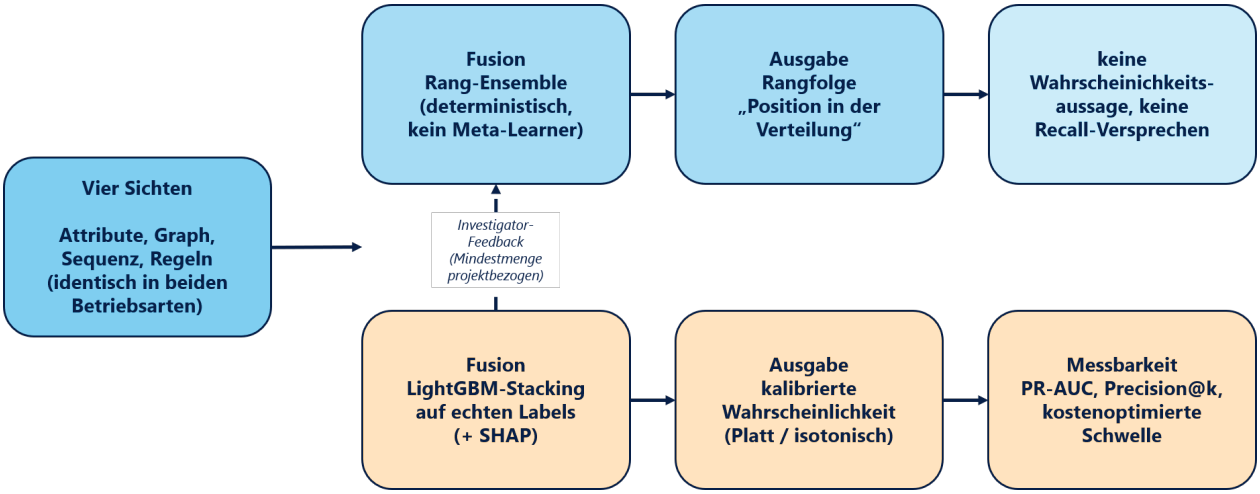

Abbildung 2: Zwei Betriebsarten, eine Pipeline. Die Verzweigung liegt ausschließlich in der Fusionsschicht; Investigator-Feedback bildet die Rampe dazwischen.

Dimension	Ohne Labels (unsupervised)	Mit Labels (supervised)
Ausgabe	Rangfolge, als Rang deklariert	Kalibrierte Wahrscheinlichkeit
Messbarkeit	Erst im Betrieb, über Prüfergebnisse	Sofort: PR-AUC, Precision@k, ECE
Stärke	Findet Muster, nach denen nie gesucht wurde	Erkennt bekannte Muster zuverlässig wieder
Typischer Fehler	Flut legitimer Ungewöhnlichkeiten	Wiederholt Ermittlungsgeschichte
Anfälligkeit	Graphkonstruktion, Saisonalität	Label-Verzerrung, Muster-Alterung
Kommunikation	„Prüfen Sie die Top 50 – mit uns”	„Die Top 50 enthalten 40 Betrüger”
SHAP/Erklärbarkeit	Feature-Beiträge zum Rekonstruktionsfehler	Vollständige Beitragsanalyse

Tabelle 1: Aufsicht und Selbstüberwachung im direkten Vergleich – Stärken, Grenzen und praktische Einsatzbedingungen.

Die letzte Zeile dieser Tabelle verdient Hervorhebung, weil sie immer wieder falsch gemacht wird. Ein unsupervisiertes Modell kann sehr wohl erklären, welche Merkmale am Rekonstruktionsfehler eines Falls beteiligt waren. Was es nicht kann, ist zu sagen, wie wahrscheinlich Betrug ist. Wer in einer Präsentation beides in einen Topf wirft, zeigt Kalibrierung ohne Kalibrierdaten.

Unser Arbeitsgrundsatz ist deshalb einfach. Wir fragen nicht „supervised oder unsupervised?“, sondern „wo auf der Rampe stehen Sie heute – und wie schnell kommen Sie weiter?” Beides mit denselben Daten und derselben Codebasis.

5 · Messen, ohne sich zu belügen

Der Teil, an dem Anomalieprojekte wirklich scheitern, ist selten das Modell, sondern meistens die Evaluation.

Zeit schlägt Zufall. Der übliche Zufallssplit – 80 Prozent Training, 20 Prozent Test – misst bei Transaktionsdaten nicht Generalisierung, sondern Gedächtnis. Betrug kommt in Wellen, Kampagnen und Ringschlüssen. Wenn der Testzeitraum denselben Ring enthält wie das Training, hat das Modell die Antwort schon gesehen. findic Anomalieerkennung teilt deshalb ausschließlich entlang der Zeitachse: Training auf der Vergangenheit, eine Embargo-Lücke, Test auf der Zukunft. Die Kennzahlen, die dabei herauskommen, sind niedriger als die aus Zufallssplits.

PR zuerst, ROC daneben. Auf Daten mit einem Prozent Positivanteil sieht die ROC-Kurve schnell besser aus, als der Betrieb ist, weil die große Masse korrekt klassifizierter Normalfälle die Falsch-Positiv-Rate drückt. Die Precision-Recall-Kurve ist bei unausgewogenen Daten daher das informativere Instrument.[19] Wir berichten PR-AUC als primäre Kennzahl und ROC-AUC als ergänzende. Dazu kommt Precision@k für die Listenlängen, die ein Ermittlungsteam realistisch abarbeitet – 50, 100, 500 Fälle. Für die Kommunikation mit Fachbereichen ergänzen wir Lift@k: die Trefferdichte der Top-k relativ zur Basisrate. „Die Top-1-%-Liste enthält achtmal so viele bestätigte Fälle wie eine Zufallsstichprobe" ist der Satz, der in einem Prüfungsausschuss ankommt, und nicht die dritte Nachkommastelle einer AUC. Eine ROC-AUC von 0,95 ist für den Betrieb nur dann hilfreich, wenn daraus auch eine brauchbare Arbeitsliste entsteht.

Kalibriert. Wahrscheinlichkeiten haben nur dann einen Wert, wenn sie gemessen halten, was sie versprechen. Gemessen wiederum wird das mit dem Expected Calibration Error (ECE). Dabei sortiert man die Fälle nach ihrem Score und teilt sie in gleich große Gruppen (Bins). In jeder Gruppe vergleicht man

[19] Saito, T. & Rehmsmeier, M. (2015), „The Precision-Recall Plot Is More Informative than the ROC Plot When Evaluating Binary Classifiers on Imbalanced Datasets", PLOS ONE 10(3): e0118432

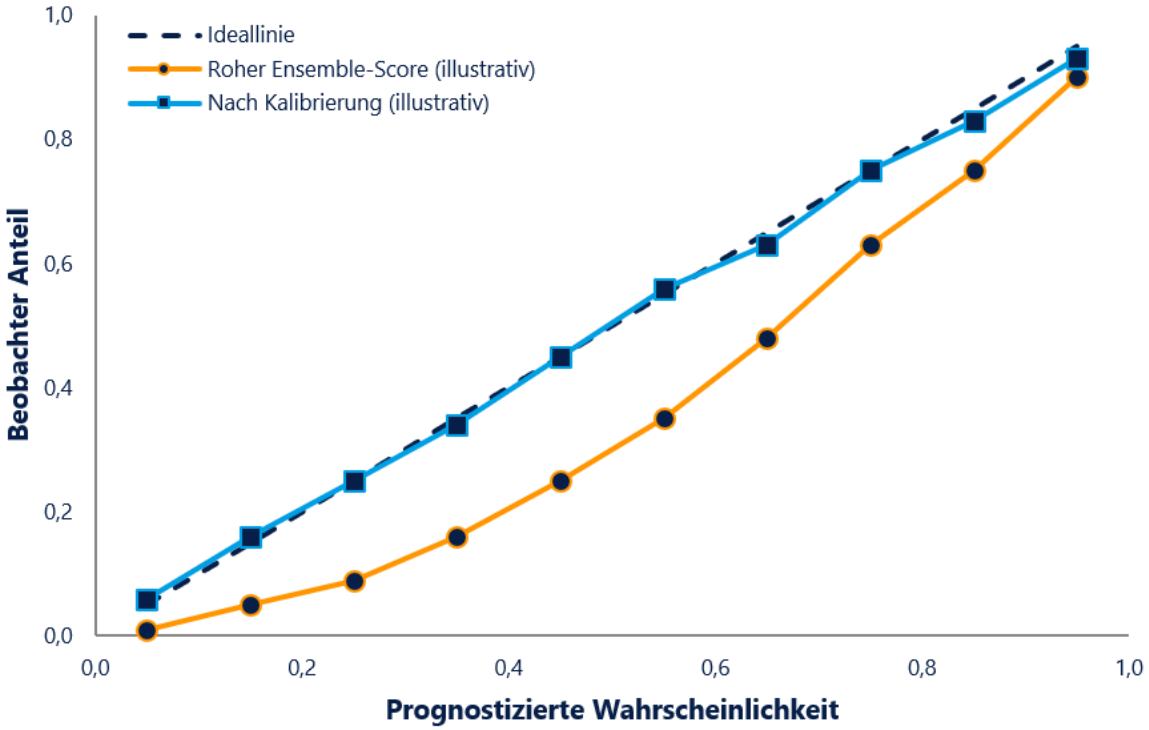

Abbildung 3: Illustratives Reliability-Diagramm. Kalibrierung ist kein Kosmetikschritt, sondern die Voraussetzung dafür, dass Schwellen und Budgets an Scores hängen dürfen. (Darstellung illustriert das Prinzip, keine Kundendaten.)

die mittlere vorhergesagte Wahrscheinlichkeit mit dem tatsächlichen Anteil bestätigter Fälle. Der ECE ist die über die Gruppen gewichtete mittlere Abweichung. Null wäre die perfekte Kalibrierung. Dabei sind moderne Netze dafür bekannt, systematisch übermütig zu sein, und ein einparametriges Nachschärfen reicht oft, um sie wieder auf Linie zu bringen.[20] Unsere Regel ist härter als der Stand der Technik. Kalibriert wird nur auf echten Labels oder auf bestätigtem Investigator-Feedback. Liegen beides nicht vor, gibt findic Anomalieerkennung Ränge aus und sagt dazu, dass es Ränge sind. Der Versuchung, aus Rangfolgen mit einer Min-Max-Streckung Scheinwahrscheinlichkeiten zwischen 0,01 und 0,99 zu gießen, widerstehen wir seit dem Tag, an dem eine solche Strecke in einem früheren Projekt in einer Managementpräsentation als „Betrugswahrscheinlichkeit" auftauchte.

Die Schwelle ist eine Controlling-Frage. Wenn ein Modell 200 Transaktionen flaggt, aber das Team 40 prüfen kann, ist die Diskussion über die dritte Nachkommastelle der AUC beendet. Wir ziehen Schwellen aus Kostenmatrizen: Was kostet eine unnötige Prüfung, was ein übersehener Schaden? Die Wahl der Schwelle leiten wir aus den Fehlerkosten und der erwarteten Basisrate ab, nicht aus einem pauschalen Default von 0,5. Dies geht auf Elkan zurück[21] und wird in der Praxis immer noch selten konsequent umgesetzt.

[20]Guo, C., Pleiss, G., Sun, Y. & Weinberger, K. Q. (2017), „On Calibration of Modern Neural Networks", *Proceedings of ICML 2017*

[21]Elkan, C. (2001), „The Foundations of Cost-Sensitive Learning", Proceedings of IJCAI 2001, S. 973–978

6 · Bewährungsproben

Ein synthetischer Stresstest – 600 generierte Transaktionen, 80 Konten, 40 injizierte Fälle mit realistischen Mustern aus beiden Deliktfeldern (hohe Beträge auf jungen Policen, Velocity-Bursts, ein Strohmann-Ring nach Geldwäsche-Muster) – dient uns als Regressionstest für jede Codeänderung. Dort trennt die Pipeline die injizierten Fälle mit einer Out-of-Time-ROC-AUC von 0,99 und einer PR-AUC von 0,96. Alle 20 Betrugsfälle des Testzeitraums stehen innerhalb der Top-50 der Risikoliste. Dabei zeigt sich, der Code funktioniert, wie er soll, über Training, Speichern, Laden und Neuscoren hinweg. Es zeigt nichts über echte Daten, auf denen Muster überlappen, Felder fehlen und Betrüger sich anpassen.

In einem Einsatz, rund 90.000 Schadenfälle, drei Jahre Historie, ohne bestätigte Labels, zeigte sich das erwartbare und trotzdem erfreuliche Bild: Die obersten 1 Prozent der Risikoliste enthielten eine sichtbare Konzentration von Fällen, die das bestehende Regelwerk als „grenzwertig, nicht eskaliert" kannte, plus eine kleine Gruppe von Fällen mit Gemeinschaftsstrukturen, die im Nachgang einer internen Prüfung einer Agentur zugeordnet werden konnten – durchgeführt von Prüfern, die die Modell-Scores nicht kannten. Zwei dieser Fälle waren bis dahin in keinem Prüfverfahren aufgefallen. Wir nennen bewusst keine Prozentzahlen. Ohne bestätigte Labels und mit einer nicht randomisierten Auswahl geprüfter Fälle wäre jede Trefferquote eine Scheingenauigkeit. Was diese Beobachtung wert ist, entscheidet sich im begleiteten Betrieb mit Feedbackschleife. Die falschen Positiven in denselben Regionen der Liste waren überwiegend echte, aber ungewöhnliche Vorgänge, wie Großeigentümer mit vielen Verträgen, also die Fehlerart, mit der wir rechnen und die sich fachlich wegargumentieren lässt.

Die nächste Ausbaustufe ist ein halbes Jahr Begleitbetrieb mit Feedbackschleife: Ermittler markieren bestätigte Treffer und Fehlalarme, und erst ab ausreichend vielen Rückmeldungen schaltet die Kalibrierung von Rang- auf Wahrscheinlichkeitsausgabe um.

7 · Wo sind die Grenzen?

Vieles davon hängt am Graphen. Die Kanten entstehen aus Schlüsseln, IDs und Adressen, und wenn die schon im Rohmaterial unzutreffend sind (Dubletten, Adressen statt Personen, Konten, die zufällig gleich benannt wurden), clustert das System Gebilde, die gar nicht existieren. Keine Modellgüte kann eine falsch verknotete Welt korrigieren. Deshalb fließt bei jedem Projekt mehr Zeit in die Graphkonstruktion als in irgendeine Modellwahl.

Ein weiterer Faktor ist die Zeit. Betrugsmuster stehen unter Anpassungsdruck: Was 2024 als Structuring auffiel, ist 2026 längst eine andere Figur, und ein Modell, das gestern trainiert wurde, erkennt die Muster von morgen nicht von allein. Anomalieerkennung ohne geplantes Retraining und Drift-Monitoring degradet.[22] Wir dokumentieren deshalb jede Modellversion mit Split-Parametern, Schwellen und Kennzahlen maschinenlesbar.

Die nächste Grenze ist konzeptioneller Art. Das System ist prädiktiv, nicht kausal: Ein hoher Score sagt „sieht aus wie die Fälle, die schlecht endeten" und nicht „ist Betrug". Wer daraus automatisierte Sperren

[22] Gama, J., Žliobaitė, I., Bifet, A., Pechenizkiy, M. & Bouchachia, A. (2014), „A Survey on Concept Drift Adaptation", ACM Computing Surveys 46(4)

ableitet statt Prüfaufträge, riskiert Entscheidungen, die fachlich wie regulatorisch schwer zu verteidigen sind. Unser System liefert priorisierte Prüflisten, und die Entscheidung bleibt beim Menschen. Das haben wir lange als Einschränkung formuliert, inzwischen sehen wir es als die einzige Betriebsart, die einer Revision standhält.

Und schließlich viertens: Substanz schlägt Architektur. Unter wenigen tausend Transaktionen oder ohne zeitliche Tiefe in der Historie messen Graph- und Sequenzsicht vor allem Rauschen. In der Situation ist die ehrliche Empfehlung nicht das komplexere Modell, sondern klassische Verfahren, sauber implementiert. Plus der Vorschlag, die Datenbasis erst aufzubauen und später wiederzukommen.

8 · Was ein Pilot bei uns bedeutet

Konkret: ein Export der Transaktions- oder Schadensdaten, wenige Wochen Laufzeit, eine fachliche Ansprechperson, die Regelwissen und Prüfkapazität kennt. Am Ende stehen eine priorisierte Risikoliste mit Begründungen pro Fall, ein schriftlicher Bericht über die gewählten Graphrelationen und Schwellenlogiken.

Die Frage, die sich nach unserer Erfahrung lohnt, ist nicht nur „Haben Sie KI?". Wichtiger ist: „Wie stellen Sie sicher, dass Ihre Evaluation belastbar ist?"

9 · Quellen / weiterführende Literatur

Chawla, N. V., Bowyer, K. W., Hall, L. O. & Kegelmeyer, W. P. (2002). SMOTE: Synthetic Minority Over-sampling Technique. *Journal of Artificial Intelligence Research* 16, 321–357. Das Referenzverfahren der synthetischen Überabtastung; interpoliert Minderheitenfälle zwischen deren nächsten Nachbarn.

Chen, T., Kornblith, S., Norouzi, M. & Hinton, G. (2020). A Simple Framework for Contrastive Learning of Visual Representations. Proceedings of ICML 2020 (PMLR 119:1597–1607). SimCLR Herkunft der NT-Xent-Verlustfunktion in unserem kontrastiven Pretraining.

Devlin, J., Chang, M.-W., Lee, K. & Toutanova, K. (2019). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. Proceedings of NAACL-HLT 2019. Das Masking-Prinzip hinter unserer Sequenzsicht.

Ding, K., Li, J., Bhanushali, R. & Liu, H. (2019). Deep Anomaly Detection on Attributed Networks. Proceedings of the 2019 SIAM International Conference on Data Mining (SDM). DOMINANT, die Blaupause des Graph-Autoencoder-Ansatzes.

Dou, Y., Liu, Z., Sun, L., Deng, Y., Peng, H. & Yu, P. S. (2020). Enhancing Graph Neural Network-based Fraud Detectors against Camouflaged Fraudsters. Proceedings of CIKM 2020. CARE-GNN. Betrugserkennung auf Graphen gegen Täter, die sich bewusst tarnen.

Elkan, C. (2001). The Foundations of Cost-Sensitive Learning. Proceedings of IJCAI 2001, 973–978. Warum die Standardschwelle 0,5 fast immer die falsche betriebswirtschaftliche Entscheidung ist.

Elkan, C. & Noto, K. (2008). Learning Classifiers from Only Positive and Unlabeled Data. Proceedings of KDD 2008. PU-Learning: der methodische Kandidat für den Umgang mit bestätigten Fällen ohne verlässliche Negative.

Fan, H., Zhang, F. & Li, Z. (2020). AnomalyDAE: Dual Autoencoder for Anomaly Detection on Attributed Networks. Proceedings of ICASSP 2020. Die sauberere Zerlegung von Attribut- und Strukturraum, an der unsere Graphsicht ausgerichtet ist.

Gama, J., Žliobaitė, I., Bifet, A., Pechenizkiy, M. & Bouchachia, A. (2014). A Survey on Concept Drift Adaptation. ACM Computing Surveys 46(4). Die Literaturbasis hinter unserem Drift-Monitoring.

Grinsztajn, L., Oyallon, E. & Varoquaux, G. (2022). Why do tree-based models still outperform deep learning on typical tabular data? NeurIPS 2022, Datasets and Benchmarks Track. Die Beleglage dafür, dass Boosting auf Tabellendaten Deep Learning schlägt.

Guo, C., Pleiss, G., Sun, Y. & Weinberger, K. Q. (2017). On Calibration of Modern Neural Networks. Proceedings of ICML 2017, 1321–1330. Übermütige Netze und die erstaunliche Wirksamkeit einfacher Nachkalibrierung.

Hilal, W., Gadsden, S. A. & Yawney, J. (2022). Financial Fraud: A Review of Anomaly Detection Techniques and Recent Advances. Expert Systems with Applications 193. Überblick über Anomalieerkennung speziell im Finanzbetrug.

Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., Ye, Q. & Liu, T.-Y. (2017). LightGBM: A Highly Efficient Gradient Boosting Decision Tree. Advances in Neural Information Processing Systems 30.

Liu, F. T., Ting, K. M. & Zhou, Z.-H. (2008). Isolation Forest. Proceedings of the 8th IEEE International Conference on Data Mining (ICDM 2008), 413–422. Das Verfahren hinter unserer Attributsicht.

Lundberg, S. M. & Lee, S.-I. (2017). A Unified Approach to Interpreting Model Predictions. Advances in Neural Information Processing Systems 30. SHAP, die Basis unserer Beitragsanalyse im supervisierten Modus.

Saito, T. & Rehmsmeier, M. (2015). The Precision-Recall Plot Is More Informative than the ROC Plot When Evaluating Binary Classifiers on Imbalanced Datasets. PLOS ONE 10(3): e0118432. Das Standardargument gegen ROC-AUC bei seltenen Positivfällen.

Sajja, B. (2026). Synthetic Tabular Generators Fail to Preserve Behavioral Fraud Patterns: A Benchmark on Temporal, Velocity, and Multi-Account Signals. arXiv:2604.13125. – Aktueller Benchmark zur Verhaltungstreue synthetischer Daten: CTGAN, TVAE, GaussianCopula und TabularARGN erreichen gute statistische Treue und brauchbare AUROC-Werte, zerstören aber die Verhaltensstrukturen, auf die Betrugserkennung angewiesen ist – Burst-Muster, Velocity-Regeln, Ringe über geteilte Geräte (Degradation um Faktor 17 bis 100 gegenüber dem realen Rauschniveau); dokumentiert zudem den Minderheitenklassen-Kollaps von TVAE, bei dem die erzeugte Betrugsrate von 3,5 % auf 0,03 % einbricht.

Tarawneh, A. S. et al. (2022). Stop Oversampling for Class Imbalance Learning: A Review. *IEEE Access*. Systematische Kritik: Synthetische Punkte fallen nachweisbar häufig in Mehrheitsregionen; Oversampling als „täuschender" Ansatz.

Veličković, P., Cucurull, G., Casanova, A., Romero, A., Liò, P. & Bengio, Y. (2018). Graph Attention Networks. ICLR 2018. Die Attention-Mechanik in unserem Graph-Encoder.

Xu, L., Skoularidou, M., Cuesta-Infante, A. & Veeramachaneni, K. (2019). Modeling Tabular Data using Conditional GAN. *NeurIPS 32*. CTGAN: das Referenzverfahren generativer Tabellensynthese.

Zadrozny, B. & Elkan, C. (2002). Transforming Classifier Scores into Accurate Multiclass Probability Estimates. Proceedings of KDD 2002. Platt-Skalierung und isotonische Regression im Vergleich.

Zheng, Y., Jin, M., Liu, Y., Chi, L., Phan, K. T. & Chen, Y.-P. P. (2021). Generative and Contrastive Self-Supervised Learning for Graph Anomaly Detection. IEEE Transactions on Knowledge and Data Engineering. SL-GAD, die Vorlage für die Kombination aus Rekonstruktion und kontrastivem Signal.

10 · Kontakt

Udo Jacobasch
Partner
Udo.Jacobasch@findic.de
Office Berlin

Dr. Josef Dirnberger
Manager
Josef.Dirnberger@findic.de
Office München

Hamburg

Kurze Mühren 20,

D-20095 Hamburg

Phone +49 40/303740-0

E-Mail info@findic.de

Zürich

Gutenbergstrasse 1,

CH-8002 Zürich

Phone +41 44/56098-00

E-Mail info@findic.ch