Diplomarbeit

Analyse von Finanzdaten mit neuronalem Data-Mining

von

Timm Sebastian Langsdorf

Betreuer: PD Dr. R. Brause

Institut für Informatik
an der
Johann Wolfgang Goethe – Universität

Frankfurt am Main,
den
28. April 1999

„... denn wenn das menschliche Gehirn so simpel wäre,
daß wir es verstehen könnten,
wären wir so simpel,
daß wir es nicht könnten.“[1]

[1] Emerson Pough

Erklärung

Hiermit erkäre ich, diese Diplomarbeit selbständig und nur mit den angegebenen Quellen und Hilfsmitteln verfaßt zu haben.

Frankfurt am Main, den 28. April 1999

Dieses Manuskript wurde mit LaTeX 2_ε gesetzt unter Verwendung von XFig 3.2 und Gnuplot 3.7

Danksagung

In erster Linie gilt mein besonderer Dank meinem Betreuer bei dieser Diplomarbeit Herrn Dr. habil. Rüdiger Brause für die Vergabe dieser aktuellen und praxisbezogenen Diplomarbeit. Weiter waren seine wertvollen Anregungen, kritischen Beiträge zu meiner Arbeit, wie auch die ausgezeichnete Zusammenarbeit mit Herrn Dr. Brause eine große Hilfe. Nicht zuletzt als ständiger Ansprechpartner zu jeder Zeit ist Herr Dr. Brause mir ein guter Begleiter bei der Diplomarbeit gewesen.

Auch danken möchte ich auf diesem Wege den Mitarbeitern der „Gesellschaft für Zahlungssysteme“ (GZS) und besonders Herrn Hepp, der mit vielen interessanten Berichten über den „Alltag“ bei der Mißbrauchsprävention mir den Bezug zur Realität geöffnet hat, und meinen Ehrgeiz mit jedem Gespräch neu entfachen konnte.

Weiter gilt der Dank meinen Eltern, die mir dieses Studium ermöglicht haben, meiner Verlobten Esther, meinem Bruder Jan wie auch der kompletten Familie Braun, die alle immer ein offenes Ohr für meine Sorgen im Zusammenhang mit der Diplomarbeit hatten, und mir geholfen haben, diese zu meistern.

Zusammenfassung

Im heutigen Zahlungsverkehr übernehmen in zunehmendem Maße Zahlungen mit Kreditkarten eine entscheidende Rolle. Entsprechend der Verbreitung dieser Art des Zahlungsverkehrs nimmt ebenfalls der Mißbrauch mit diesem bargeldlosen Zahlungsmittel zu. Um die Verluste, die bei dem Kreditkarteninstitut auf diese Weise entstehen, so weit wie möglich einzudämmen, wird versucht, Mißbrauchstransaktionen bei der Autorisierung der Zahlungsaufforderung zu erkennen.
Ziel dieser Diplomarbeit ist es zu bestimmen, in wie weit es möglich ist, illegale Transaktionen aus der Menge von Autorisierungsanfragen mit Hilfe adaptiver Algorithmen aufzudecken. Dabei sollen sowohl Methoden aus dem Bereich des Data-Mining, als auch aus den Bereichen der neuronalen Netze benutzt werden.

Erschwerend bei der Mißbrauchsanalyse kommt hinzu, daß die Beurteilung der einzelnen Transaktionen in Sekundenbruchteilen abgeschlossen sein muß, um die hohe Anzahl an Autorisierungsanfragen verarbeiten zu können und den Kundenservice auf Seiten des Benutzers und des Händlers auf diese Weise zu optimieren. Weiter handelt es sich bei einem Großteil der bei der Analyse zu Verfügung stehenden Datensätze um symbolische Daten, also alpha-numerisch kodierte Werte, die stellvertretend für verschiedene Eigenschaften verwendet werden. Nur wenige der Transaktionsdaten sind analoger Natur, weisen also eine Linearität auf, die es erlaubt, „Nachbarschaften“ zwischen den Daten bestimmen zu können. Damit scheidet eine reine Analyse auf Basis von neuronalen Netzwerken aus. Diese Problematik führte unter anderem zu dem verfolgten Ansatz.
Als Grundlage der Analyse dienen bekannte Mißbrauchstransaktionen aus einem Zeitintervall von ungefähr einem Jahr, die jedoch aufgrund der hohen Anzahl nicht komplett als solche mit den eingehenden Transaktionen verglichen werden können, da ein sequentieller Vergleich zu viel Zeit in Anspruch nähme. Im übrigen würde durch einen einfachen Vergleich nur der schon bekannte Mißbrauch erkannt werden; eine Abstraktion der Erkenntnisse aus den Mißbrauchserfahrungen ist nicht möglich. Aus diesem Grund werden diese Mißbrauchstransaktionen mit Hilfe von Methoden aus dem Bereich des Data-Mining verallgemeinert und damit auf ein Minimum, soweit es die Verläßlichkeit dieser Datensätze zuläßt, reduziert. Desweiteren schließt sich eine Analyse der zu diesem Zeitpunkt noch nicht betrachteten analogen Daten an, um die maximale, enthaltene Information aus den Transaktionsdaten zu beziehen. Dafür werden moderne Methoden aus dem Bereich der neuronalen Netzwerke, sogenannte radiale Basisfunktionsnetze, verwendet.
Da eine Mißbrauchsanalyse ohne eine entsprechende Profilanalyse unvollständig wäre, wurde abschließend mit den vorhanden Mitteln auf den zugrunde liegenden Daten in Anlehnung an die bisherige Methodik eine solche Profilauswertung und zeitabhängige Analyse realisiert. Mit dem so implementierten Modell wurde versucht, auf allgemeine Art und Weise, Verhaltens- beziehungsweise Transak-

tionsmuster einzuordnen und mit bei der Mißbrauchsentscheidung einfließen zu lassen.

Aus den vorgestellten Analyseverfahren wurden verschiedene Klassifizierungsmodelle entwickelt, die zu guten Ergebnissen auf den Simulationsdaten führen. Es kann gezeigt werden, daß die Mißbrauchserkennung durch eine kombinierte Anwendung aus symbolischer und analoger Auswertung bestmöglich durchzuführen ist.

Inhaltsverzeichnis

Konventionen

M_j	Mißbrauchsart j mit $j = 1, \ldots, p$
M_0	legale Transaktion
R_{ij}	Regel bestehend aus Eingabetupel $\mathbf{x_i} = x_{i1}, \ldots, x_{in}$ und der Klassenzuordnung M_j
$(\hat{R})_{ij}$	verallgemeinerte, aus Regel R_{ij} generalisierte Regel
r_{ij}	Anzahl der Regeln $R_{ij} = \|R_{ij}\|$
$\hat{r}_{ij}$	Anzahl der von Regel $\hat{R}_{ij}$ abgedeckten Regeln (im Fall einer Generalisierung durch Wildcards)
$A_j(\mathbf{x}_i)$	Assoziation auf Eingabemuster $\mathbf{x_i}$
X	Regelrumpf (Prämisse)
Y	Regelkopf (Konklusion)
$X \longrightarrow Y$	Regel, Assoziation von X und Y
$\mathcal{D}$	komplette Menge aller Transaktionen (legal sowie illegal)
$\mathcal{M}$	komplette Menge aller Mißbrauchstransaktionen
α	Ähnlichkeitsmaß unter den Regeln, gibt die Anzahl der Unterscheidungen an
S_i^{Layer}	Squashingfunktion – Funktion, die auf die Eingabe des Neurons i in der Schicht *Layer* angewandt wird
B_i	Basisfunktion i
$B_i(\mathbf{x})$	Ausgabefunktion des i-ten RBF-Neurons
s	Schwellwert oder Bias
E	Quadratischer Fehler
$\mathbf{c}$	Zentrumsvektor der radialen Basisfunktion bzw. des RBF-Neurons
$\mathbf{x}$	Eingabevektor
γ	Lernrate
σ	Varianz (Ausdehnung) der Glockenfunktion
$\mathbb{R}$	Raum der reellen Zahlen
$\mathbb{R}^n$	n-dimensionaler Raum der reellen Zahlen
$P(x)$	Wahrscheinlichkeit (*engl.:* prior probability) von x
$P(x\|y)$	bedingte Wahrscheinlichkeit (*engl.:* conditional probability) von x unter Bedingung y
S	Wahrscheinlichkeitsraum, dient als Grundlage für die Bestimmung der Auftrittswahrscheinlichkeit von symbolischen Datentupeln
p_t^{NN}	*richtige* Mißbrauchsprognose aufgrund der Analyse mit Hilfe der neuronalen Netze
p_f^{NN}	*falsche* Mißbrauchsprognose aufgrund der Analyse mit Hilfe der neuronalen Netze

$H(t)$	Entropie des Datums t
U	Zustandsraum, Basis einer Markov-Kette mit den Zuständen $U = \{u_1, \ldots, u_J\}$ mit den Zufallsvariablen $\{x_1, \ldots, x_n\}$
$\|.\|$	Betrag
#	Anzahl
GL	Generalisierungslevel
Support	prozentualer Anteil der Regel im Datenbereich
Konfidenz	bedingte Wahrscheinlichkeit für das Auftreten der Regel
Abdeckung	Wahrscheinlichkeit für das Auftreten in der speziellen Klasse

Kapitel 1

Einführung und Überblick

1.1 Gliederung der Arbeit

Einleitend in diese Diplomarbeit soll nun eine grobe Gliederung der einzelnen Kapitel vorangestellt werden. Auf diese Weise soll die Orientierung in der Arbeit und den einzelnen Kapiteln erleichtert und überschaubarer gemacht werden.
Zunächst soll mit dem folgenden Abschnitt 1.2 in die genaue Problematik, die dieser Diplomarbeit zugrunde liegt, eingeführt werden. Darauf folgt in Anschluß an dieses Kapitel in Kapitel 2 eine Beschreibung der vorliegenden Daten sowie die genaue Vorgehensweise, diese in eine arbeitstaugliche Form zu bringen und Transaktionsdaten und Karteninhaberdaten zueinander in Beziehung zu setzen. Zusätzlich sei hier auf den Anhang B verwiesen, in dem statistische Eigenschaften der einzelnen Felder aufgeführt sind, sowie auf Anhang A, der eine Auflistung sämtlicher vorliegender Daten inklusive einer kurzen Beschreibung enthält.
In Kapitel 3 folgt die Entwicklung eines Ansatzes zur Auswertung der symbolischen Werte in den Transaktionsdatensätzen. Dazu wird zunächst das Modell des Assoziativspeichers vorgestellt, das Schritt für Schritt mit Hilfe von Methoden und Vorgehensweisen aus der Statistik beziehungsweise des Data-Minings für die vorliegende Problematik optimiert wird. Desweiteren ist ein Abschnitt der Implementierung des entwickelten Verfahrens gewidmet, und es sei schon an dieser Stelle auf den verwendeten Algorithmus in Pseudokode in Anhang C verwiesen. Daran anschließend folgt eine Auswertung der Ergebnisse, die mit diesem Modellansatz erzielt werden konnten.
Im folgenden Kapitel 4 wird auf Basis von radialen Basisfunktionsnetzen eine Analyse der Analogdaten vorgestellt. Es handelt sich dabei um eine Mehrschichtige Netzarchitektur, die die vorhandenen Analogdaten aus verschiedenen Blickwinkeln auswertet. Die erwartungsgemäß nicht ausreichende Erkennungsrate allein dieser Analysemethode wird anhand der aufgeführten Ergebnisse deutlich. Da auf die Technik der radialen Basisfunktionsnetze ausführlich eingegangen wird, sei bis auf die im Anhang D näher beschriebene Klassenhierarchie des implementier-

ten RBF-Netzmodells nicht weiter auf die eigentliche algorithmische Umsetzung eingegangen und auf die beigefügte HTML-Dokumentation der Implementierung verwiesen.
In Verbindung mit der Mißbrauchsanalyse kann eine Profilauswertung helfen, anhand von Zeitsequenzen und -reihen die Klassenentscheidung zu verbessern. Aus diesem Grund wird in Kapitel 5 eine spezielle Profilanalyse für die vorliegende Problematik entwickelt und diskutiert. Dabei wird sowohl auf die symbolischen als auch auf die analogen Daten zurückgegriffen. Eine Beschreibung der verwendeten Verfahren findet sich ab Abschnitt 5.3. Die Implementierung lehnt sich stark an die der radialen Basisfunktionsnetze aus Kapitel 4 an und soll ebenfalls nur eingehend durch die beigefügte HTML-Dokumentation kommentiert werden. Desweiteren sind die Ergebnisse, die mit Hilfe dieser Profilauswertung auf den allgemeinen Datenmengen erzielt werden konnten, aufgeführt. Es schließt sich eine Auswertung dieser Ergebnisse an.
Nachdem nun verschiedene Analyseverfahren für die einzelnen Datentypen vorgestellt sind, wird darauf aufbauend in Kapitel 6 anhand verschiedener Präventionsmodelle versucht, eine akzeptable Lösung für das vorliegende Problem zu finden. In Zusammenhang mit den einzelnen Modellansätzen werden die jeweils erzielten Ergebnisse vorgestellt. Abschließend sollen diese dann in Abschnitt 6.5 zusammenfassend in Relation zueinander gesetzt, ausgewertet und diskutiert werden.
Abschließend fügt sich das Kapitel 7 an, in dem Möglichkeiten aber auch Grenzen des vorgestellten Systems aufgezeigt werden; aber auch ein Ausblick auf eventuelle Verbesserungsmaßnahmen wird gegeben.

1.2 Die Ursache des Problems

Im heutigen Zahlungsverkehr werden vermehrt Kreditkarten eingesetzt. Der bargeldlose Handel nimmt auch im Privatbereich weltweit zu. Doch mit diesem Wachstum wächst auch das Problem des zunehmenden Mißbrauchs mit dieser Art des Zahlungsmittels.

Es ist nun notwendig diesen Mißbrauch so weit wie möglich einzudämmen, indem man auf Basis bekannter, als Mißbrauchstransaktion verifizierter Transaktionsdaten Vergleiche zu den aktuellen Kreditanfragen herstellt, und diese aufgrund der gemachten Erfahrung beurteilt.
Schon seit Jahren werden in diesem Zusammenhang neuronale Netze eingesetzt, die aber nicht ausreichend präzise, geschweige denn nachvollziehbar, die eintreffenden Transaktionen auswerten. Weiter werden zur Mißbrauchsprävention Expertensysteme und allgemeine „black-box“-Netzwerke eingesetzt, die mit Hilfe von Spezialisten auf diesem Gebiet auf dem aktuellen Stand gehalten werden müssen. Dies zeigt unter anderem, daß die Mißbrauchseigenschaften nicht statisch sind, also ein System, das zur Prävention eingesetzt werden soll, variabel

und selbstständig auf die jeweiligen Änderungen und Wandlungen des Mißbrauchs reagieren muß.

Im folgenden soll daher zunächst eine möglichst exakte Problembeschreibung gegeben werden, um darauf aufbauend ein neuartiges System zur Mißbrauchsprävention zu entwickeln.

1.3 Die Präventionsproblematik

Hauptproblem bei der Mißbrauchsprävention ist die eigentliche Erkennung einer Mißbrauchstransaktion aus der großen Anzahl an zu verarbeitenden Transaktionsdaten. Erschwerend kommt jedoch zu dem eigentlichen Problem der Mißbrauchserkennung hinzu, daß nur sehr wenig Zeit zur Autorisierung einer Kreditanfrage zur Verfügung steht. Allein die Anzahl der täglich bei dem Kreditinstitut eintreffenden Zahlungsanfragen in Höhe von 400000 Kreditanweisungen – Tendenz steigend – läßt einen ausgiebigen Vergleich mit bekannten Mißbrauchstransaktionen nicht zu. Umgerechnet auf die Sekunde macht das 4,5 eintreffende Transaktionen. Hochgerechnet auf ein Jahr sind das 132 Millionen Buchungsaufträge. Die Tatsache des Mißbrauchs und des hohen Durchsatzes macht eine kontrollierende, überwachende Verarbeitung in „fast“ Echtzeit notwendig; die Autorisierung einer Zahlungsanforderung muß unmittelbar nach dem Eintreffen erfolgen.

Eine weitere Schwierigkeit ist die im Verhältnis zu der Gesamtanzahl an zu verarbeitenden Transaktionen geringe Anzahl an Mißbräuchen. Die Wahrscheinlichkeit eines Mißbrauchs liegt nach Schätzungen von Experten bei etwa 0,2%. Dieser Anteil kann jedoch durch die bestehenden Systeme auf einen Prozentsatz von 0,1% gemindert werden. Dennoch verschuldet dieser geringe Anteil an illegalen Transaktionen bedeutenden, finanziellen Schaden. Eine rigorosere Mißbrauchsprävention jedoch verursacht bei Falscheinstufungen von zulässigen Kreditanfragen Unzufriedenheit beim Kunden, die im schlimmsten Fall zu einer Kundenemigration führt. Außerdem ist der Gewinnverlust durch eine verweigerte Kreditdienstleistung auch finanziell zu berücksichtigen. Desweiteren erzeugt jeder Fehlalarm auch einen direkten finanziellen Schaden, da dieser über ein bestimmtes Verfahren mit einer „Strafgebühr“ abgegolten werden muß.

Kommt es zu einem Kreditkartenbetrug, zum Beispiel durch „Cloning“, Diebstahl oder Manipulation an Karte oder Terminal, so liegt der Schaden direkt bei dem Kreditkarteninstitut. Das Kreditinistiut haftet für den Kunden, der Opfer des Mißbrauchs geworden ist. Insofern ist das Interesse des Kreditkarteninstitutes an einer optimalen Mißbrauchsabwehr, ohne jedoch überempfindlich und damit zuviele Fehlalarme auszulösen, nachvollziehbar.

An dieser Stelle sei auch schon auf einen Punkt hingewiesen, der zwar unmittelbar auf das Problem der Mißbrauchserkennung keinen Einfluß hat, aber dennoch im Rahmen dieser Diplomarbeit beachtet werden muß. So ist es beim derzeitigen Stand nicht oder nur sehr schwer möglich, die von dem aktuell verwendeten System abgewiesenen, tatsächlichen Mißbräuche protokollieren zu können. Aus diesem Grunde liegen dieser Diplomarbeit nur die Mißbrauchstransaktionen zugrunde, die erst im Nachhinein als Mißbrauch deklariert werden konnten. Das heißt, als Grundlage dieser Diplomarbeit dienen Daten, die mit den derzeit verwendeten Analysemethoden und Präventionssystemen, schon verarbeitet und zum großen Teil[1] von Mißbrauch bereinigt sind.
Im folgenden sei noch einmal das Problem anhand der wesentlichen Eckpunkte charakterisiert:

Schnelle Autorisierung: Bei 4,5 Autorisierungsanfragen in der Sekunde muß ein Prüfverfahren fast in Echtzeit zu einer Entscheidung kommen. Hinzu kommt, daß eine schnelle Abwicklung zum Kundenservice zählt.

Geringer Prozentsatz an Mißbrauch: Der vermutete Mißbrauchsanteil liegt laut Kreditinstitut bei 0,2%. Nur bei einem Tausendstel der autorisierten Transaktionen (0,1%) handelt es sich um einen nicht verhinderten, erst im Nachhinein erkannten Mißbrauch. Dieses Faktum der unausgewogenen Datenverteilung der legalen und illegalen Transaktionen gestaltet ein Training neuronaler Netze bezüglich der bekannten Mißbrauchsdaten sehr schwierig. Außerdem ist nicht bekannt, in welchem Umfang Mißbrauch derzeit von dem aktuellen System auch als solcher erkannt und abgewiesen wurde, um auch auf diese Art von Mißbrauch weiter eingehen zu können. Es ist nur in Ausnahmefällen bekannt, ob es sich bei einer abgewiesenen Autorisierungsanfrage um einen tatsächlichen Mißbrauch gehandelt hat.
Zuletzt sei angemerkt, daß einem neuronalen Netz, und sei es noch so gut trainiert, es **nicht** möglich ist, eine 99,9% Trefferquote bei der Klassifizierung, wie sie letztendlich durch das Transaktionsverhältnis von 1:1000 gegeben ist, zu erreichen, wenn es zu Überlappungen bezüglich der zu unterscheidenden Klassen kommt.

Hoher Anteil an Symboldaten: In den Transaktionsdatensätzen inklusive der Karteninhaberdaten ist ein hoher Anteil an symbolischen Werten enthalten. Dabei handelt es sich um verschlüsselte Werte, Kodierungen oder Identifikationsnummern, die zum Beispiel Lokalitäten, Währungen oder transaktionsspezifische Eigenschaften repräsentieren. Von den insgesamt 38 Daten enthalten 26 symbolische Werte. Also sind 68% der Daten in einer Transaktion symbolischer Natur. Nur wenige Daten enthalten analoge Zahlenwerte,

[1] Nach Aussage des Kreditinstituts kann die Mißbrauchsquote von 0,2% auf 0,1% vermindert werden (siehe oben)

wie zum Beispiel die Felder Transaktionsbetrag, Transaktionsdatum oder Kreditlimit.

Hohe Streuung der Symboldaten: Der Wertebereich der symbolischen Daten umfaßt zum Teil mehr als 100.000 verschiedene Werte. Im Anhang B sind diese Eckdaten zu den einzelnen Datenfeldern für die vorliegenden Daten aufgeführt.

Keine Mißbrauchshistorie: Es ist derzeit nicht möglich, auf Basis der Transaktionsdaten auf eine Transaktionshistorie zurückgreifen zu können. Ein solches, zurückverfolgendes Zeitfenster muß jeweils speziell aus der Datenbank abgefragt werden.

Wenige Informationen: In den eigentlichen Transaktionsdaten sind nur wenige Zusatzinformationen enthalten, die nicht unmittelbar die eigentliche Transaktion betreffen, wie zum Beispiel die genaue Lokalität, an der die Transaktion getätigt wurde, oder die Frequentierung der Kreditkarte. Solche Informationen könnten helfen, die einzelnen Anfragen genauer einzustufen. Statt dessen ist man unter Umständen darauf angewiesen, aus der Währung Rückschlüsse auf das Land in dem die Bezahlung stattfindet zu ziehen. Die einzelnen zur Verfügung stehenden Daten sind in Anhang A aufgeführt. Einige Daten tauchen sowohl in den Karteninhaberdaten als auch in den Transaktionsdaten auf.

Verschiedene Mißbrauchsarten: Institutsintern werden 13 verschiedene Mißbrauchsarten unterschieden, auf die unterschiedlich reagiert werden kann. Es ist jedoch bei der Mißbrauchsanalyse zunächst ohne Bedeutung, welche Mißbrauchsart genau vorliegt, so daß nur zwischen legaler und illegaler Autorisierungsanfrage zu entscheiden ist.

Vertrauensgrenze: Laut Aussage der GZS ist es zunächst akzeptabel, eine vorliegende Transaktion, die mit einer Zuverlässigkeit von 10% als Mißbrauchsregel zugewiesen werden kann, auch als Mißbrauch zu definieren. Das heißt, taucht ein Transaktionsmuster in 10 von 100 Fällen tatsächlich in illegalen Autorisierungsanfragen auf, so kann mit ausreichender Sicherheit jede weitere damit übereinstimmende Transaktion als Mißbrauchstransaktion eingestuft werden.

Vorangegangene Mißbrauchsanalyse: Die der Arbeit zugrundeliegenden Daten sind nach den zum Zeitpunkt der Transaktionen verwendeten Analyse- und Präventionsmaßnahmen auf Mißbrauch hin untersucht worden. Die abgewehrten, potentiellen Mißbrauchsversuche fallen somit aus der Menge der zu analysierenden Daten heraus, das heißt ein Großteil der tatsächlichen Mißbrauchsversuche wurde im voraus erkannt und verhindert und taucht somit *nicht* in dem verwendeten Datenmaterial auf.

Durch das unausgewogene Verhältnis aus legalen und illegalen Autorisierungsanfragen, wie auch aufgrund des häufigen Auftretens der symbolischen Werte ist eine alleinige Verarbeitung durch ein klassisches, neuronales Netz nicht möglich. Auf der anderen Seite kann eine Einordnung, allein aufgrund der symbolischen Werte, genauer zu realisieren sein, als mit den wenigen, analogen Daten.
Für eine realitätsnahe Umsetzung der Problemstellung ist es notwendig, die aufgeführten Umstände zu berücksichtigen und bei der Simulation auf den vorliegenden Daten zu beachten.

Kapitel 2

Datenaufbereitung

2.1 Motivation

Als Grundlage für die Analyseverfahren, die mit dieser Diplomarbeit vorgestellt werden, dient dank der Mitarbeit der „Gesellschaft für Zahlungssysteme“ (GZS) reales Datenmaterial aus dem Kreditkartensektor. Bei den Daten handelt es sich um orginale Transaktions- und Kontendaten, bei denen aus datenschutzrechtlichen Gründen die Kontonummer, der POS-Code[1], die Mißbrauchsart sowie die Kontonummer eineindeutig verschlüsselt wurden.
Die in ASCII-Format gelieferten Rohdaten konnten jedoch nicht in der ursprünglichen Form verwendet werden, da die einzelnen Datensätze erst von zum Beispiel Verwaltungsdatensätzen und ungültigen Buchungen, sowie Fremdkartentransaktionen, bereinigt werden mußten. Weiter sollte ein Datenszenario geschaffen werden, das es erlaubte, die Analyse der Daten so nahe wie möglich an der Realität durchzuführen. Dazu mußten unter anderem die Daten zum Beispiel in zeitlich geordnete Reihenfolge umsortiert werden.

In diesem Abschnitt soll deswegen näher auf die Anordnung beziehungsweise Struktur der Daten eingegangen und Zusammenhänge erörtert werden. Abschnitt 2.2 befaßt sich darum kurz mit dem hierarchischen Aufbau der Daten und den gegebenen Zusammenhängen. In Abschnitt 2.3 wird anhand von SQL-Klauseln die Filterung und Formatierung der Rohdaten erklärt und grob auf die Statistiken der resultierenden Daten eingegangen.

[1] **P**oint **O**f **S**ale: Ort, an dem die Transaktion getätigt wurde (siehe auch im Anhang A)

2.2 Datenstruktur

2.2.1 Die Datenbasis

Wie in der Motivation im vorherigen Abschnitt 2.1 erwähnt, ist ein Teil der Daten aufgrund datenschutzrechtlicher Bestimmungen durch alpha-numerische Werte unkenntlich gemacht worden. Dabei wurde besonderer Wert darauf gelegt, daß diese Verschlüsselung *eineindeutig* ist, um die Datenanalyse sinnvoll gestalten zu können. Es ergibt sich also aus der Verschlüsselung heraus kein Problem bei der Verarbeitung, so daß kein Informationsverlust gegenüber den Orginaldaten gegeben ist.
Außerdem muß gewährleistet sein, daß es sich bei den vorliegenden Daten um eine möglichst repräsentative Auswahl der Transaktionen handelt. Dies ist in sofern gewährleistet, als es sich um Transaktionen aus ungefähr $1\frac{1}{4}$ Jahren (siehe Anhang B) handelt. Es muß weiter sichergestellt sein, daß die Daten in ihrer vorliegenden Form mit den orginalen, eintreffenden Transaktions- beziehungsweise vorliegenden Kundendaten übereinstimmen, da es sonst keinen Sinn macht, eine zuverlässige Auswertung auf Basis dieser Daten durchzuführen.

Abbildung 2.1: Datenhierarchie

Ursprünglich liegen die Daten in vier Tabellen, wie auch in Abbildung 2.1 gezeigt, vor:

- `TRX`: Sämtliche Transaktionsdaten inklusive der 0,1% Mißbrauchstransaktionen, soweit eine Transaktion nicht schon zum Zeitpunkt der Autorisierungsanfrage als Mißbrauch erkannt und abgewehrt wurde. Dabei handelt es sich um eine grobe, aber dennoch repräsentative Auswahl aus allen getätigten Transaktionen über den in Anhang B aufgeführten Zeitraum.

- `TRX_MISS`: Die Mißbrauchstransaktionen, bei denen erst nach einer erfolgten Autorisierung ein Mißbrauch festgestellt werden konnte.

- `KIAKTIV`: Die aktuellen, aktiven Kunden- beziehungsweise Kontendaten.

- `KISPERR`: Die aktuellen, gesperrten Kunden- beziehungsweise Kontendaten.

Die Anordnung der Datensätze ist mit dem internen Ablauf der Transaktionsverarbeitung bei der GZS vorgegeben. Darum soll nun schematisch ein Ablauf einer solchen Anfrage zusammenfassend dargestellt werden:

1. Die GZS erreicht eine Zahlungsanforderung. Je nach Länderbestimmung gibt es einen Betragsschwellenwert, der eine Autorisierung erst notwendig macht. Kleinere Beträge werden ohne eine besondere Prüfung, ohne Autorisierung, gewährt. Die Betragsschwelle ist von Land zu Land unterschiedlich.

2. Ist eine Autorisierung notwendig, wird diese in Form einer Überprüfung durch die vorhandenen Mißbrauch erkennenden Systeme ausgeführt.

3. Gilt die Kreditanforderung als legal, wird sie unter den Transaktionen (`TRX`) verbucht.

4. Unter Umständen stellt sich später heraus, daß ein Mißbrauch stattgefunden hat. Dies kann einige Tage bis Wochen dauern, je nach Kundenreaktion und Mißbrauchsart. Zu den eigentlichen Transaktionsdaten werden in diesem Zusammenhang zusätzliche Informationen entsprechend Tabelle A.2 in Anhang A bezogen, die jedoch *nicht* für die eigentlichen, präventorischen Maßnahmen zum Zeitpunkt der Transaktion zur Verfügung stehen.

5. Auf jeden Fall findet nach der Abbuchung vom Kundenkonto ein sogenanntes *Posting* statt, bei dem die Transaktion zusammenfassend, kundenabhängig gesammelt und weiter protokollierend verarbeitet wird.

6. Im Falle eines Mißbrauchs ist die GZS „Schuldner“, das heißt, sie muß für den unrechtmäßig verbuchten Betrag einstehen. Das Geld wird in jedem Falle dem betrogenen Händler ausgezahlt.

2.2.2 Datenanalyse

Die zu verarbeitenden Daten bestehen aus den bei der Transaktion registrierten Daten und den lokal gespeicherten Konteninhaberdaten. Es stehen also insgesamt 38 Daten, wie im Anhang A aufgeführt, zur Analyse einer Transaktion zur Verfügung. Diese setzten sich aus den 19 eigentlichen Transaktionsdaten und 19 Kontoinhaberdaten zusammen, die sich unter Umständen zum Teil sogar überschneiden wie zum Beispiel im Falle des Kreditlimits oder natürlich der Kontonummer, die als Schlüssel[2] fungiert.
Der Vollständigkeit halber wurde den Transaktionen die entsprechende Mißbrauchsart in Form eines kodierten Wertes angehängt[3]. Diese Mißbrauchsart

[2]Der sog. „Erstschlüssel“ bzw. „primary key“, eindeutiges Kennzeichen eines Datensatzes

[3]Bei den legalen Datensätzen wurde das Mißbrauchsfeld nicht gefüllt sondern als `NULL` gesetzt.

konnte aus der Tabelle TRX_MISS entnommen werden. An dieser Stelle sei noch einmal erwähnt, daß die Informationen, die ausschließlich in der Tabelle TRX_MISS abgelegt sind, *nicht* zum Zeitpunkt der eigenlichen Transaktion zur Verfügung stehen. Stattdessen werden diese speziellen Auskünfte erst eingeholt, sobald eine Transaktion als Mißbrauchstransaktion erkannt wird. Aus diesem Grund wird diese Datenbasis nur zur rückführenden Bestimmung der eigentlichen Mißbrauchstransaktionen benutzt. Anhand Graphik 2.1 ist diese Datenkorrelation nachzuvollziehen.

2.2.3 Datensatzanalyse

Bei der Analyse der Testdaten fällt schnell auf, daß es sich bei den Daten eines kompletten Transaktionsdatensatzes im wesentlichen mehr um symbolische als um analoge Werte handelt. 26 der 38 ($\approx 68\%$) Daten enthalten verschlüsselte beziehungsweise kodierte Werte für die verschiedenen Transaktionseigenschaften, wie zum Beispiel Firmencodes, Kartenart oder Währung, in Form einer alphanumerischen Zeichenkette. Die einzelnen Datenfelder können dabei bis zu mehrere tausend Werte annehmen wie im Anhang B, Tabelle B.2, deutlich wird.

2.3 Datenbereinigung und -formatierung

Wie weiter oben schon erwähnt, sind unter anderem in den Transaktionsdatensätzen (TRX) Fremdkartentransaktionen verzeichnet, die im Feld CTY_1 mit AIRPLUS gekennzeichnet sind. Außerdem sind in den Daten Transaktionen aufgeführt, die keine Autorisierung beabsichtigten. Darunter fallen zum Beispiel akute Kartensperrungen, Diagnoseanfragen und Konfigurationsveränderungen oder auch die oben zitierten Postings. Diese sind für das vorliegende Problem der Mißbrauchserkennung nicht von Bedeutung und brauchen ebenfalls nicht weiter betrachtet zu werden.
Die Transaktionsdatensätze (TRX) entsprechen dem Format der eingehenden Daten bei der GZS. Weitere Zusatzinformationen, wie sie zum Beispiel in den Mißbrauchsdaten (TRX_MISS) enthalten sind, werden erst im Nachhinein, wenn die Transaktion intern weiterverarbeitet wird oder sich als Mißbrauch herausstellt, erhoben. Auf diese Daten kann also **nicht** bei der Analyse zur Autorisierung zugegriffen werden.

2.3.1 Generierung verwertbarer Mißbrauchsdaten

Um nun komplette Mißbrauchstransaktionen generieren zu können, die als Basis für die simulierte Analyse der Mißbrauchsdaten dienen sollen, müssen die in den Mißbrauchsdaten (TRX_MISS) enthaltenen Transaktionen den entsprechenden, vollständigen Datensätzen aus den eigentlichen Transaktionsdaten (TRX) zu-

geordnet werden. Dafür wurden zunächst folgende Kriterien festgelegt, die mit den dokumentierten SQL-WHERE-Klauseln[4] erzielt wurden.

- Der Transaktionszeitpunkt ist in den Mißbrauchsdaten `TRX_MISS` nur als Datum aufgeführt. Eine genaue Uhrzeit liegt nicht vor. Demzufolge muß eine Übereinstimmung der ersten 10 Stellen, dem Datum, ausreichen.

  ...WHERE LEFT([TRX].[TRN_DT],10)=LEFT([TRX_MISS].[TRN_DT],10)...

- Desweiteren wurde auf Übereinstimmung des `ICA_CD` und des `SIC_CD` getestet. Es handelt sich dabei um den Transferpartner (zumindest im außereuropäischen Raum) und das Branchenkennzeichen.

  ...WHERE [TRX].[ICA_CD]=[TRX_MISS].[ICA_CD] AND
  [TRX].[SIC_CD]=[TRX_MISS].[SIC_CD]...

- Der Transaktionsbetrag kann nicht direkt verglichen werden, da bis zur Erkennung des Mißbrauchs ein unterschiedlich langer Zeitraum liegt, in dem Kursschwankungen zu Abweichungen führen. Um rückführend die Transaktionen bezüglich dieses Datums zuordenen zu können, wird (laut GZS) von einer Abweichung von $\pm 20\%$ bezüglich der Mißbrauchsbetragswerte ausgegangen. Diese Abweichungen von $\pm 20\%$ werden in die Abfrage mit aufgenommen.

  ...WHERE WERT([TRX].[TRN_AMT])/100 BETWEEN (
  [TRX_MISS].[TRN_AMT]−([TRX_MISS].[TRN_AMT]*20/100) AND
  [TRX_MISS].[TRN_AMT]+([TRX_MISS].[TRN_AMT]*20/100)
)...

- Ein weiterer wichtiger Faktor ist die Verknüpfung der Transaktionsdaten (`TRX` und `TRX_MISS`) mit den Kontendaten über die Kontonummer (`ACCT_NBR`). Hierzu wurden die Kontonummern der Transaktionsdaten mit denen der Kontendaten „gejoint“. Das heißt, es wurden nur die Transaktionsdatensätze betrachtet, die ein Äquivalent der Kontonummer in den gesperrten Kontendaten enthalten. Nur die Transaktionsdatensätze, in denen die Kontonummer übereinstimmend in den beteiligten Tabellen auftauchte, wurden ausgewählt. Zu den übrigen sind demnach *keine* Karteninhaberdaten in der vorliegenden Datenmenge `KISPERR` vorhanden.

  ...FROM (KISPERR INNER JOIN TRX_MISS ON
  [KISPERR].[ACCT_NBR] = [TRX_MISS].[ACCT_NBR])
  INNER JOIN TRX ON
  [KISPERR].[ACCT_NBR] = [TRX].[ACCT_NBR]...

[4] *engl.*:**S**tructured **Q**uery **L**anguage: Sprache zur Definition und Manipulation relationaler Datenbanken. SQL wurde in den 70er Jahren von IBM entwickelt und ist mittlerweile unabhängiger Standard[Ea93]

Es werden alle Datensätze berücksichtigt, die in den angegebenen Daten – in diesem Fall den Kontonummern – übereinstimmen.

- Zu guter Letzt wurden die Datensätze entfernt, bei denen das Feld `CTY_1` mit dem Wert `AIRPLUS` gefüllt war. Hierbei handelt es sich, wie erwähnt, um Karten von Fremdkreditinstituten, die über das Transaktionsnetz der GZS abgewickelt werden, hier aber nicht von Bedeutung sind.

  ...WHERE KISPERR.CTY_1<>"AIRPLUS"...

- Eine Sortierung der Datensätze wurde bezüglich des Datums und der Uhrzeit vorgenommen, um spätere Zugriffe zu vereinfachen und die Daten in einer realistische Reihenfolge anzuordnen.

Bei diesen Verknüpfungen konnten insgesamt 8885 von den insgesamt 117901 Mißbrauchstransaktionsdatensätzen zu vollständigen Datensätzen zusammengefügt werden. Hierbei ist zu beachten, daß es sich bei den Datensätzen in `TRX` *nur* um eine Auswahl aller getätigten Transaktionen in dem angegebenen Zeitraum (siehe Anhang B). Die auf diese Weise erhaltenen Transaktionsdaten entsprechen nun in der Datenzusammensetzung den in der Realität bei dem Kreditinstitut eintreffenden Transaktionsdaten. Durch die vage Zuweisung des Transaktionsbetrags (`TRX_AMNT`) durch die Abweichung von ± 20% des Mißbrauchs-Transaktionsbetrags kommt es in manchen Fällen zu doppelten Zuweisungen ein und desselben Transaktionsdatensatzes. Diese doppelten und damit identischen Mißbrauchstransaktionen werden nun mit folgender SQL-Abfrage entfernt, so daß letztendlich 5850 Mißbrauchstransaktionen übrig bleiben.

SELECT DISTINCT TRX_KENNUMMER * FROM TRX;

Die so bearbeiteten Daten werden abschließend in der Tabelle `ILLEGALDAT` abgelegt und gespeichert.

2.3.2 Bereinigung der legalen Datensätze

Auch die legalen Transaktionsdatensätze mußten nach bestimmten Kriterien gefiltert werden. Es gilt die ungültigen Transaktionen, wie Anfragen, Kontensperrungen etc., auszusortieren, damit ausschließlich nur Autorisierungsanfragen zur Weiterverarbeitung übrig bleiben. Folgend die Auswahlkriterien für die legalen Transaktionen mit den zugehörigen Filterfunktionen in SQL-Form:

- Zunächst wurden auch hier die Transaktionen der Karten von fremden Kreditinstituten herausgefiltert.

  ...WHERE KIAKTIV.CTY_1<>"AIRPLUS"...

- Weiter wurden Transaktionen, die keine Kreditanforderung enthielten, entfernt. Dies konnte mittels des Transaktionstyps festgestellt werden. Dieser `TRN_TYP`[5] enthält kodiert an 1. Stelle den sogenannten „Acquiring Typ“, der für die Auswertung uninteressant ist, und somit nicht weiter betrachtet wird. Die 2. Stelle enthält die „finanzielle Bedeutung“, die unter anderem Limitveränderung = `X` und Kartensperrung = `Y` beinhaltet. Die 3. Stelle ist der „TRX- Bezug“, der auch ein Storno- Flag `S` beinhaltet, das für die Datenverarbeitung zur Mißbrauchserkennung irrelevant ist.

  ...WHERE(
  MID([TRX].[TRN_TYP],2,1)<>“K“ OR
  MID([TRX].[TRN_TYP],2,1)<>“X“ OR
  MID([TRX].[TRN_TYP],2,1)<>“Y“)
  AND
  MID([TRX].[TRN_TYP],3,1)<>“S“...

  Die übrigen Werte in diesem Zusammenhang, die die einzelnen Flags beinhalten können, tauchen nicht auf oder brauchen nicht weiter berücksichtigt werden.

- Eine Verknüpfung zwischen den Transaktionsdaten und den aktiven Kontendaten sorgt auch hier dafür, daß nur *die* Transaktionen übrigbleiben, die ein Pendant in den aktiven Kontendaten enthalten.

  ...FROM KIAKTIV INNER JOIN TRX ON
  [KIAKTIV].[ACCT_NBR] = [TRX].[ACCT_NBR]...

- Auch hier wurde wieder eine Sortierung nach dem Transaktionsdatum vorgenommen.

Durch die Filterung der Transaktionsdatensätze nach obigen Kriterien konnten von den ursprünglich 713873 Datensätzen, in denen ebenfalls die Mißbrauchstransaktionen enthalten sind, insgesamt 542858 Transaktionsdatensätze aktiven Konten zugeordnet werden. Bei diesen Transaktionsdaten kann davon ausgegangen werden, daß es sich um ausschließlich legale Transaktionen handelt. Diese werden in der Tabelle `LEGALDAT` gespeichert.

2.3.3 Vervollständigung der legalen Analysedaten

Um sicherzugehen, daß ausschließlich vollständige Daten vorliegen, das heißt keine Transaktionen **ohne** zugehörigen Karteninhaber oder umgekehrt existieren, wurden abschließend die Daten dahingehend weiter bearbeitet.

[5] siehe auch Beschreibung `TRN_TYP` in Anhang A

Nach den bisher beschriebenen Datenselektionen wurden die legalen Transaktionen ausgewählt, zu denen aktive, beziehungsweise im Falle von illegalen Transaktionen gesperrte Konten vorrätig waren. Aus diesem Grund wird nun die Menge der Karteninhaberdaten ebenfalls derart eingeschränkt, daß nur noch die Konten übrig bleiben, zu denen auch entsprechende Transaktionen existieren.

```
SELECT DISTINCT KIAKTIV.* INTO KIAKTIVTRX FROM KIAKTIV
    INNER JOIN LEGALDAT
        ON [KIAKTIV].[ACCT_NBR] = [LEGALDAT].[ACCT_NBR];
```

Damit enthält die Tabelle `KIAKTIVTRX` sämtliche kontenspezifischen Daten (jeweils einmal) aus dem Datensatz der aktiven Kontendaten, zu denen in den bisher bearbeiteten und selektierten legalen Transaktionsdaten eine Transaktion vorliegt. Damit wurde die Anzahl der ursprünglich vorliegenden 59005 aktiven Konten auf eine Zahl von 42782 Konten mit vorhandenen Transaktionen reduziert.

Entsprechend den aktiven Konten wurden nun auch die gesperrten Konten selektiert, zu denen Mißbrauchstransaktionen vorliegen. Hierzu wurde auf die zuvor gefilterten Mißbrauchstransaktionsdaten `ILLEGALDAT` zurückgegriffen:

```
SELECT DISTINCTROW KISPERR.* INTO KISPERRTRX FROM KISPERR
    INNER JOIN ILLEGALDAT
        ON [KISPERR].[ACCT_NBR] = [ILLEGALDAT].[ACCT_NBR];
```

Somit existiert in der Tabelle `KISPERRTRX` zu jeder Mißbrauchstransaktion ein gesperrtes Konto und umgekehrt. Von den vorher bekannten 14643 gesperrten Konten konnten damit nur noch 1120 Konten vollständige Transaktionen zugeordnet werden.

2.4 Ergebnisse der Datenbereinigung

Nach der Datenbereinigung entsprechend der vorhergegangenen Abschnitte dezimierten sich die Daten auf die in Tabelle 2.1 aufgeführten Quantitäten der Datensätze.

Bezeichnung	**Beschreibung**	**Anzahl Datensätze**
`KIAKTIVTRX`	aktive Kontendatensätze	42.782
`KISPERRTRX`	gesperrte Kontendatensätze	1.120
`LEGALDAT`	legale Transaktionsdatensätze	542.858
`ILLEGALDAT`	illegale Transaktionsdatensätze	5.850

Tabelle 2.1: Anzahl der arbeitsfähigen Datensätze

Die Ergebnisse der statistischen Auswertungen der vorliegenden Daten sind im Anhang B aufgeführt.

2.5 Die Datenverwaltung

In den vorhergegangenen Abschnitten wird ohne explizite Erwähnung die Datenformatierung und -bereinigung mit Hilfe einer relationalen Datenbank auf Basis von SQL-Abfragen ausgeführt. Schon hier wird die Hilfe durch solch eine Datenbank ersichtlich; es braucht so keine umständliche Textsuche über den kompletten Datenbestand programmiert und ausgeführt werden.
Auf diese Weise wird auf die optimierten Datenbankalgorithmen zurückgegriffen, um komplizierte Abfragen auf die Daten anzuwenden. Im wesentlichen kommt es so zu einer Vereinfachung der Datenpflege und -verwaltung sowie einem übersichtlichen und strukturierten Aufbau der Daten.
Die Verwendung von JAVA in Verbindung mit einer solchen Datenbank wird durch die "Java Database Connectivity" – kurz JDBC – Schnittstelle ermöglicht, mit der es einfach zu realisieren ist, die Datenbank über SQL-Kommandos zu steuern und abzufragen.
Nicht zuletzt die spontane Überprüfung verschiedener datenbasierter Eigenschaften und Ergebnisse konnten auf diese Weise einfach und effizient realisiert und ausgeführt werden, ohne für die verschiedenen Aufgaben spezielle Abfragekonstrukte fehleranfällig programmieren zu müssen.

Kapitel 3

Verarbeitung der symbolischen Daten mit Data-Mining-Verfahren

Wie nun mehrfach darauf hingewiesen, sind die Transaktionsdaten derart aufgebaut, daß sie zum Großteil aus symbolischen Daten bestehen. Vorrangiges Ziel muß es also sein, aus einer Menge von Datensätzen möglichst viele Mißbrauchstransaktionen mit Hilfe von Mustern aus verschiedenen, symbolischen Werten herauszufiltern und einem bestimmten Mißbrauchstypus zuzuordnen. Es handelt sich demnach im Prinzip um ein Zuordnungsproblem, das an eine konventionelle Art der fehlertoleranten Mustererkennung erinnert.

Es bietet sich das Modell des Assoziativspeichers an, das in den nächsten Abschnitten näher beschrieben wird. Weiter soll versucht werden, Lösungen für die Nachteile, die sich in diesem Zusammenhang mit diesem Modell ergeben, zu entwickeln. Dabei wird auf Methoden des Data-Mining Bezug genommen – zum einen um eine theoretische Grundlage für die verwendeten Algorithmen zu schaffen und zum anderen um auf die Begriffswelt aus diesem Bereich zurückgreifen zu können.

3.1 Motivation

Bei der Mißbrauchsprävention ist es von großer Bedeutung, illegale Transaktionen so schnell wie möglich aufzudecken und zu erkennen. Dabei soll eine solche Mißbrauchseinstufung jedoch mit ausreichender Sicherheit durchgeführt werden, damit es nicht zu Fehlalarmen kommt. Hinzu kommt, daß eine solche Erkennung nur sehr wenig Zeit in Anspruch nehmen darf und deswegen eine schnelle Methode für die Filterung gefunden werden muß.
Beschränkt man sich auf die Analyse der symbolischen Daten, so ergeben sich

Wertemuster, die einer speziellen Mißbrauchsart zugeordnet werden können. Die Möglichkeit, jeweils den gesamten Musterraum der bekannten Mißbrauchstransaktionen zu durchsuchen, kann auf der einen Seite aus zeittechnischen Gründen nicht realisiert werden, auf der anderen Seite fehlt jegliche Fehlertoleranz gegenüber unwesentlichen Abweichungen zwischen den Mißbrauchsmustern aus den symbolischen Werten. Die Möglichkeit der Verallgemeinerung beziehungsweise der Abstraktion wird dabei nicht berücksichtigt.
Neuronale Assoziativspeicher haben mehrere Vorteile gegenüber normalen Speichermodellen. Im Gegensatz zur sequentiellen Suche in einem Listenspeicher mit inhaltsorientierter Adressierung wird ein Zugriff auf die möglichst effektiv gespeicherten Muster mit inhaltsorientierten Schlüsselworten realisiert [Bra95]. Weiter bietet eines der bekanntesten Assoziativspeichermodelle, der Korrelations-Matrixspeicher, die Möglichkeit einer fehlertoleranten Assoziation.
Auf diese Weise können in Bezug zum vorliegenden Problem die Tupel aus symbolischen Transaktionsdaten mit den entsprechenden Mißbrauchstypen assoziiert werden. Durch Berücksichtigung der datumsspezifischen Eigenarten können Vergleichsoperationen optimiert, statistische Wertigkeiten erkannt und ausgenutzt sowie durch zusätzliche Verfahren der Eingaberaum verallgemeinert und damit unter Umständen reduziert werden.

In diesem Kapitel soll, nachdem der Begriff der Entropie eingeführt ist, näher auf das Modell des Assoziativspeichers eingegangen werden (Abschnitt 3.3). Die Nachteile, die dieses Modell mit sich bringt, sollen aufgezeigt werden (Abschnitt 3.3.2) und durch eine Weiterentwicklung des Speichermodells in Abschnitt 3.4 mit einer zusätzlichen, stochastischen Schicht behoben werden. Die Grundlagen für die wahrscheinlichkeitsbedingte Verarbeitung aus dem Bereich des Data-Mining und die spätere Realisierung des Assoziativspeichers wird in Abschnitt 3.5 eingehend beschrieben. In Abschnitt 3.9 wird dann der algorithmische Aufbau des implementierten Assoziativspeichers skizziert und anhand von Beispielen erörtert. Abschließend werden auszugsweise die erzielten Ergebnisse vorgestellt (Abschnitt 3.12) und ausgewertet (Abschnitt 3.12.3).

3.2 Die Entropie als Informationsmaß

Bevor mit der Auswertung und Analyse der Daten begonnen wird, sollen zunächst die zugrundeliegenden Datentypen und Werte näher betrachtet und analysiert werden. Eine Möglichkeit ist, die Daten auf ihre Varianz hin zu untersuchen. Da es sich jedoch um symbolische Daten handelt, muß ausweichend auf das Merkmal der Entropie zurückgegriffen werden, die in diesem Zusammenhang ein geeignetes Maß für diese Art von symbolischen Daten ist. Die Entropie erlaubt es Rückschlüsse bezüglich der Verteilung beziehungsweise Streuung der symbolischen Daten eines Merkmals zu beurteilen.

Die Entropie $H(t)$ einer Quelle oder eines Datums t entspricht der erwarteten Information pro Zeichen beziehungsweise pro Einheit. Es ist ein quantitatives Maß zur Bestimmung des Informationsgehalts basierend auf den Wahrscheinlichkeiten der einzelnen Informationseinheiten. Im folgenden sei eine Herleitung des Entropiewertes in Anlehnung an Brause [Bra95] Seite 57 ff. nachvollzogen.

Jede Information kann auf das Niveau einer binär kodierten Zahl transformiert werden. Mit n Bits können zum Beispiel 2^n Informationen in Form von Zahlen kodiert werden. Also ist die Information proportional zu n, der Anzahl von Bits:

$$I \sim n = \mathrm{ld}(2^n) \tag{3.1}$$

Sind alle verschiedenen Speicherzustände und damit Informationen in Form von Zahlen gleichwahrscheinlich, also ist für eine endliche Menge an Ereignissen (im vorliegenden Fall sind das die Zahlen) eine gleiche Wahrscheinlichkeit P von $\frac{1}{2^n}$ gegeben, dann gilt für die Information eines solchen Zustandes (Zahl):

$$I \sim \mathrm{ld}(\frac{1}{P}) \quad \text{mit} \quad P = \frac{1}{2^n} \tag{3.2}$$

Die in dem Ereignis enthaltene Information kann aber auch in Form eines natürlichen Logarithmus mit Hilfe des Transformationsmoduls [BSMM93] ausgedrückt werden durch:

$$\ln(.) = \ln(2) \cdot \log_2(.) \tag{3.3}$$

Nun kann speziell für ein Ereignis t_k (zum Beispiel eine Zahl) die Information I mit $I(t_k)$ angegeben werden. Das Ereignis t_k habe die Wahrscheinlichkeit $P(t_k)$.

$$I(t_k) := ln\left(\frac{1}{P(t_k)}\right) = -ln(P(t_k)) \tag{3.4}$$

Nebenbei sei hier erwähnt, daß die Information zweier unabhängiger Ereignisse die Summe der Informationen der Einzelereignisse ist.
Wenn ein Ereignis t_k also fast sicher ist, das heißt $P(t_k) \longrightarrow 1$, geht der Betrag der darin enthaltenen Informationen gegen Null. Ist hingegen die Wahrscheinlichkeit eines Ereignisses sehr klein, das heißt $P(t_k) \longrightarrow 0$, ist die im Ereignis enthaltene Information sehr hoch.
Die mittlere Information von N Ereignissen, also t_k für $k = \{1, \ldots, N\}$ mit **gleicher** Wahrscheinlichkeit $P = \frac{1}{N}$, ermöglicht es, die Entropie in Form des Mittelwertes zu berechnen.

$$H(t) := \frac{1}{N} \sum_{k=1}^{N} I(t_k) = \sum_{k=1}^{N} \frac{1}{N} I(t_k) \tag{3.5}$$

Diese *durchschnittliche* Information nennt man auch *Entropie* $H(T)$, wobei T eine Zufallsvariable mit den möglichen Ergebnissen t_k ist. Treten die t_k allerdings mit **ungleichen** Wahrscheinlichkeiten auf, so müssen die tatsächlichen Wahrscheinlichkeiten $P(t_k)$ verwendet werden. Anstelle der „mittleren Information“ handelt es sich damit dann um die erwartete Information, gegeben durch den jeweiligen Erwartungswert $\langle . \rangle$.

$$H(t) := \sum_{k=1}^{N} P(t_k) I(t_k) = \sum_{k=1}^{N} P(t_k) \cdot -\ln(P(t_k)) = \langle I(t_k) \rangle_k \tag{3.6}$$

Es ist $\langle f(t_k) \rangle_k := \sum_k P(t_k) f(t_k)$ der Erwartungswert von f.
Es gilt, $H(t)$ erreicht ein Maximum, wenn alle Ereignisse gleich „wahrscheinlich“ sind, das heißt $t_k = \frac{1}{N}$ für $k = 1, 2, \ldots, N$, und ein Minimum, wenn alle Ereignisse bekannt sind.
Die Entropie $H(t)$ einer Nachrichtenquelle t entspricht also der erwarteten Information pro Zeichen einer Nachrichtenquelle.
In Anlehnung an [Pat97] zeigt Bild 3.1 den Verlauf der Entropie für den Fall zweier Ereignisse. Dabei gilt für das Ereignis t_2 der Zufallswert $P(t_2) = 1 - P(t_1)$. Damit erreicht die Entropie ein Maximum bei einer ausgewogenen Verteilung mit $P(t_1) = P(t_2) = 0,5$.

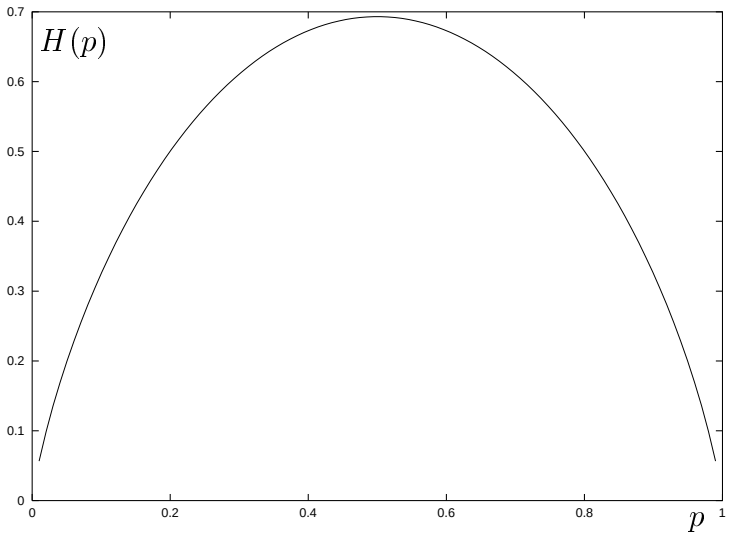


Abbildung 3.1: Entropie als Funktion der Wahrscheinlichkeit eines Ereignisses t_k für $N = 2$

In Bezug zu den symbolischen Datenwerten kann nun mit Hilfe der Entropieberechnungen eine Einstufung des Informationsgehaltes für jedes symbolische Datum errechnet werden. Die Ergebnisse dieser Berechnungen sind in Anhang E zusammenfassend dargestellt. Ein hoher Entropiewert bedeutet in diesem Zusammenhang, daß von einer hohen Varianz der Werte ausgegangen werden kann

und somit die Werte nur wenig aussagekräftige Informationen – auch in Bezug zu einer eindeutigen Mißbrauchszuweisung – enthalten. Auch kann davon ausgegangen werden, daß vermehrt unterschiedliche Ausprägungen des Merkmals existieren und diesbezüglich keine prägnanten Werte zu erkennen sind. Ist die Entropie eines Datums dagegen klein, so kann davon ausgegangen werden, daß nur wenige verschiedene Werte existieren, als auch nur vereinzelte Werte bezüglich dieses Merkmals vorherrschen.

3.3 Der binäre Assoziativspeicher

3.3.1 Theorie und Aufbau des Assoziativspeichers

Aufgrund der Vorteile von Assoziativspeichern soll ein die Mißbrauchstransaktionen klassifizierendes Analysemodell mit Hilfe dieser Art neuronaler Netze realisiert werden. Dabei werden zunächst ausschließlich die symbolischen Daten betrachtet.
Die Assoziativspeicher können in drei nicht-disjunkte Klassen eingeteilt werden [Roj93]:

Heteroassoziativer Speicher: Bei dieser Art des Assoziativspeichers werden Eingabevektoren $\mathbf{x}_i$ mit Ausgabevektoren $\mathbf{y}_i$ in Verbindung gebracht, wobei die Dimensionen der Vektoren verschieden sein können. Einem Eingabemuster $\mathbf{x}_i$ oder einem, das diesem ähnlich ist, wird das Ausgabemuster $\mathbf{y}_i$ zugeordnet. Wird die Anzahl der zu speichernden Vektorpaare zu groß in Relation zu den verwendeten Neuronen, so versagt dieses Modell.

Autoassoziativer Speicher: In diesem Fall werden die Vektoren $\mathbf{x}_i$ mit sich selbst assoziiert, das heißt in Bezug zum heteroassoziativen Speicher ist $\mathbf{y}_i = \mathbf{x}_i$. Dieses Modell wird dazu benutzt, verrauschte oder unvollständige Eingaben zu vervollständigen.

Mustererkennungsnetz: Hierbei handelt es sich um einen Spezialfall der heteroassoziativen Netze, bei denen statt eines Ausgabevektors ein Skalar mit der Eingabe $\mathbf{x}_i$ assoziiert wird.

Für das vorliegende Problem bietet sich demzufolge das Mustererkennungsnetz an. Doch zunächst soll das Modell des Korrelations-Matrixspeichers eingehender beschrieben werden.

Das Eingabemuster sei durch den reellen Eingabevektor $\mathbf{x} = (x_1, \ldots, x_n)^T$ beschrieben [1], die damit verknüpften Ausgabemuster durch $\mathbf{y} = (y_1, \ldots, y_m)^T$; man spricht auch von dem Assoziationstupel A. Auf Seite 24 in Abbildung 3.2 ist ein

[1] Das hochgesetzte „T“ bezeichnet den transponierten Vektor

Modell eines Assoziativspiechers visualisiert. Die Assoziation beider Muster läßt sich mittels der Matrix $\mathbf{W} = w_{ij}$ angeben, so daß sich Gleichung 3.7 ergibt. Die Gewichte w_{ij} bewerten die Verknüpfung der einzelnen, assoziierten Musterpaare.

$$\mathbf{y} = \mathbf{W}\mathbf{x} \tag{3.7}$$

Diese Gewichtsmatrix $\mathbf{W}$ wird in der Regel mit der Hebb'schen Lernregel trainiert, und es ergibt sich für w_{ij} bei einem Musterpaar (x_i, y'_j), wobei y'_j im Falle des Trainings der gewünschten Lehrervorgabe, dem „Sollwert" bezüglich x_i, entspricht:

$$\Delta w_{ij} = \gamma_t x_i y'_j \tag{3.8}$$

Dabei bezeichnet γ_t die aktuelle Lernrate zum Zeitpunkt t . Sollen mehrere (k) Muster gespeichert werden, so gilt:

$$w_{ij} = \sum_k \Delta w_{ij} = \sum_k \gamma_k x_i^k y_j^k \qquad \text{mit initialem Gewicht} \qquad w_{ij}(0) := 0 \tag{3.9}$$

und für die Gesamtgewichtsmatrix:

$$\mathbf{W} = \sum_k \gamma_k \mathbf{x}^k \mathbf{y}^k \tag{3.10}$$

Sind die Gewichte gelernt, so kann mit Hilfe der Gewichtsmatrix $\mathbf{W}$ das Assoziationstupel $A = \mathbf{y}$ wie in Gleichung 3.7 bestimmt werden.

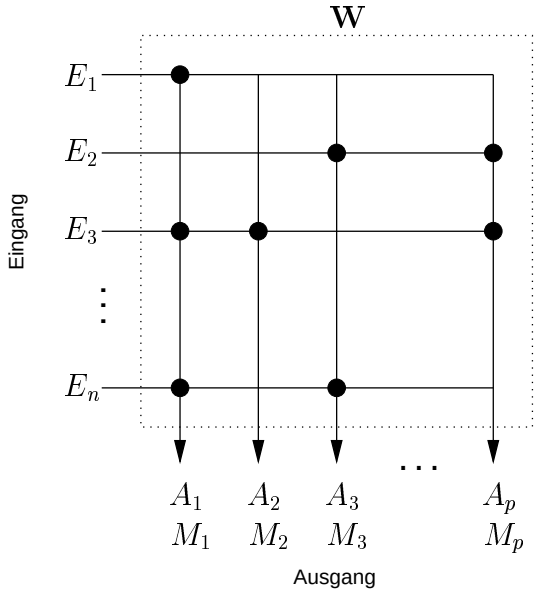


Abbildung 3.2: Modell eines Assoziativspeichers

Durch das Übersprechen[2] von anderen präsentierten Eingabemustern oder durch leichte Abweichungen vom ursprünglichen Muster erhält man das gespeicherte Ausgabemuster und einen kleineren „Störterm“, der aus einer Linearkombination aller gespeicherten Ausgabemuster $\mathbf{y}^k$ gebildet wird.
Durch Hinzunahme einer Schwellenwertoperation kann die Ausgabe derart modifiziert werden, daß die Störterme nicht zur Geltung kommen und somit nur die gewünschte Assoziation ausgegeben wird. Der Schwellenwert s_j läßt sich als ein Toleranzmaß gegenüber Abweichungen in den Eingabedaten auffassen. Bei geeigneter Schwelle werden nur die stärksten Komponenten diese Schwelle überschreiten, das „Rauschen“, also die Störungen, haben keine ausreichende Wirkung auf die Ausgabe. Als Beispiel für eine solche Schwellenwertfunktion sei Gleichung 3.11 gegeben:

$$A_j = \begin{cases} 1 & y_j \geq s_j \\ 0 & y_j < s_j \end{cases} \quad \text{wobei} \quad y_j = \sum_i w_{ij} x_i \quad \text{und} \quad s_j = \sum_{k=1}^{n} w_{kj} \tag{3.11}$$

Dabei kann die Schwelle s_j wie in Gleichung 3.11 mit Gewichten aus der Menge $w_{ij} \in \{0, 1\}$ als Summe aller vorhandenen Gewichte des Ausgabeneurons gewählt werden.

> Als Beispiel dienen folgende Szenarien, die in Anlehnung an Abbildung 3.2 beschrieben werden sollen. Zunächst seien die Eingabevariablen E_2 und E_n gesetzt. Damit ergibt sich ausschließlich mit den als Punkten gesetzten Gewichten $w_{ij} \in \{0, 1\}$ die Ausgabe A_3, da hier die Schwelle von 2 erreicht wird. In einem anderen Szenario sind die Eingaben E_2 und E_3 gesetzt. In diesem Fall kommt es zu der Ausgabe von A_2 *und* A_p. Dies führt also zu einer Mehrdeutigkeit bei der Klassifizierung.

3.3.2 Problemdiskussion

Wie im Beispiel zu sehen, gerät man jedoch schnell an die Grenzen eines solchen Speichers, wenn versucht wird, eine größere Menge an Mustern zu speichern. Besonders das Faktum, daß in der Realität nicht wie unter Optimalbedingungen die Eingabemustervektoren orthogonal zueinander ausfallen, führt mit zunehmender Musteranzahl zu Mehrdeutigkeiten, wie im Beispiel, oder sogar zu Falschzuweisungen bei der Ausgabe des Speichers. Auch die Wahl der Schwelle kann dieses Problem nicht allgemein beheben. Da es nur einen Schwellenwert gibt, kann die Schwellenwertentscheidung nicht gleichzeitig allen Klassengrenzen gerecht werden; deswegen ist es nur möglich, einen Schwellenwert zu finden, der einen möglichst geringen Klassifizierungsfehler mit sich bringt. Jede Klassengrenze zu anderen Klassen verlangt von einem Neuron einen besonderen

[2] *engl.*: crosstalk

Schwellenwert[Bra95].

Eine Alternative, die verschiedenen Muster orthogonal zu konzipieren, ist, alle symbolischen Werte eindeutig mit binären Werten zu kodieren, so daß für jeden Wert eines Datums eine neue Eingabevariable angelegt wird, die dann einzig auf 1 gesetzt ist. Anschaulich ist dies in Abbildung 3.3 gezeigt.

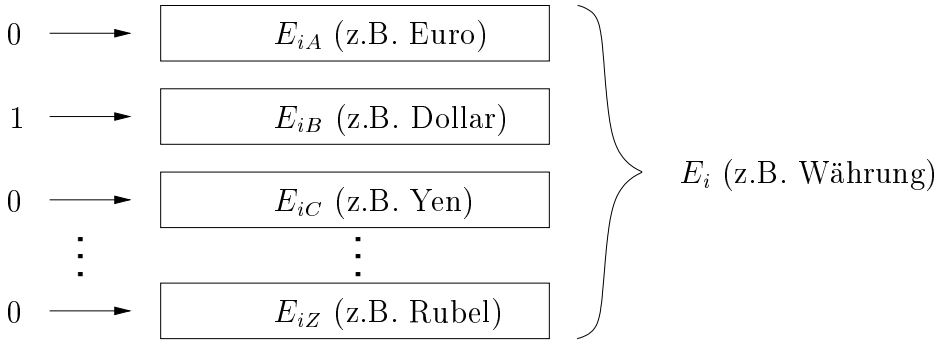


Abbildung 3.3: Mögliche Kodierung der symbolischen Werte

Dieser Ansatz der Kodierung der Eingabe hat jedoch wie bei vielen Assoziativnetzen zur Folge, daß der Eingaberaum nur sehr „dünn“ mit Informationen besetzt ist, da sehr viele Eingänge auf 0 gesetzt sind. Dies wiederum hat zur Folge, daß der Assoziativspeicher – die Matrix $\mathbf{W} = w_{ij}$ – ebenfalls nur sehr *dünn*[3] besetzt ist, also unnötig viel Platz in Relation zur Informationsdichte verbraucht. Bei dem vorliegenden Problem bedeutet dies genau, daß bei der großen Datenfülle fast nichts in der Speichermatrix $\mathbf{W}$ gespeichert ist, also nur sehr wenige der Gewichte gesetzt sind[4].

Für die Umsetzung der Transaktionszuordnung treten weitere Probleme auf, die der Verwendung des dargestellten Assoziativspeichers entgegenstehen. Zum einen ist ein *Verlernen* einmal antrainierter Muster nicht vorgesehen, so daß veraltete Regeln eine Zuordnung unnötig erschweren und „verrauschen“. Nur ein „Überlernen“ ist möglich, bei dem zuvor trainierte Muster mit neuen Assoziationen verbunden werden können und die vorhandenen Gewichte überlagert werden. Zum anderen ist das Lernen der Gewichte von verschiedenen Parametern wie Lernrate und Initialgewicht abhängig, die, zunächst ohne Bezug zu den einzuordnenden Werten, benutzerseitig gewählt werden müssen.

Zusammenfassend also noch einmal die Nachteile eines klassischen Assoziativspeichers für die vorliegende Problemstellung:

[3] *engl.*: sparse

[4] An dieser Stelle sei trotz der Untauglichkeit der Realisierung auf diese Weise die Schwelle entsprechend des Beispiels 3.11 mit $s_j = n$ angegeben, wobei n der Anzahl der einzelnen, symbolischen Datentypen entspricht.

1. Mehrdeutigkeiten bei der Zuordnung der Mißbrauchsart sind möglich
2. hoher Speicherverbrauch durch dünn besetzte Gewichtsmatrix aufgrund vieler verschiedener Werte
3. kein Verlernen möglich, nur ein „Überlernen“
4. abhängig von verschiedenen, benutzerdefinierten Parametern

Im folgenden soll nun versucht werden, ein Modell zu entwickeln, bei dem diese Nachteile behoben sind und das damit für die Mißbrauchsanalyse bezüglich der symbolischen Transaktionsdaten geeignet ist.

3.4 Erweitertes Assoziativspeichermodell

Wie im vorhergehenden Abschnitt beschrieben, kann es zu Mehrdeutigkeiten bei der Zuordnung kommen, die sich gerade bei der Klassifizierung in nur zwei Klassen – Mißbrauch oder nicht Mißbrauch – entscheidend auswirken können.
Es bedarf also einer „assoziationsbeurteilenden“ Schicht, die die Assoziationen zusätzlich zu der Klassenentscheidung beeinflußt. Ein mögliches Modell ist in Abbildung 3.4 skizziert. Wie dargestellt, ist außerdem denkbar, eine dritte Schicht anzufügen, die eine binäre Klassenentscheidung in „legal“ oder „illegal“ vornimmt. Bei der *Schicht 1* handelt es sich um eine einfache Assoziativschicht, die

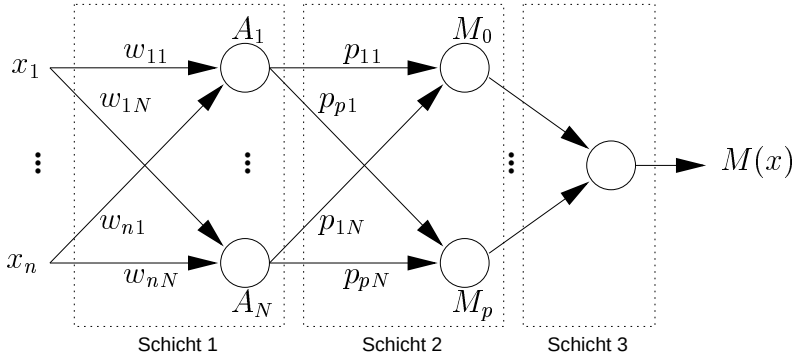


Abbildung 3.4: Modell eines Netzwerks zur Mißbrauchserkennung

zum Eingabevektor das Assoziativtupel A_i bestimmt. Hier werden die Eingaben entsprechend der gelernten beziehungsweise antrainierten Gewichte w_{ij} verarbeitet und mit den entsprechenden Ausgaben – Assoziationen A_i – assoziiert. Diese werden dann in der *Schicht 2* auf Basis der Auftrittswahrscheinlichkeiten der Assoziationstupel ausgewertet und an die letzte *Schicht 3* weitergereicht, wo eine binäre Klassenentscheidung mit Hilfe einer Schwellenwertoperation getroffen wird.

3.4.1 Organisation der Assoziativschicht

Für eine spätere wahrscheinlichkeitsgestützte Analyse der Assoziationspaare ist es sinnvoll, die Datentupel $\mathbf{x}_i = \{x_{i1}, \ldots, x_{in}\}$ aus den symbolischen Daten samt ihrer Klassenzuweisung M_j zu einem Paar zusammenzufassen. Dieses soll entsprechend der Konventionen in der Warenkorbanalyse[5][Bol96] als *Regel* R_{ij} bezeichnet werden. Auf den Begriff *Regel* soll in Abschnitt 3.5 ausführlicher eingegangen werden. Mit Hinblick auf das Assoziationstupel gilt:

$$\{M_j, x_{i1}, \ldots, x_{in}\} = R_{ij} \equiv (A_j(\mathbf{x}_i) = 1 \Rightarrow M_j = 1) \tag{3.12}$$

wobei M_j mit $j = 1, \ldots, p$ die entsprechende Mißbrauchsart bezeichnet. Die legalen Transaktionen beziehungsweise Datentupel sollen von nun an mit M_0 assoziert werden, so daß die Regelmenge R_{ij} mit $\{R_{ij}|j = 0, \ldots, p\}$ nun *auch* die legalen Datentupel umfaßt. Mit $A_j(\mathbf{x})$ wird die Ausgabe, also die Assoziation, des j-ten Neurons in der 1. Schicht bezeichnet. Ist dieses Neuron aktiviert, wird vorerst von der Mißbrauchsart M_j ausgegangen.

Aus Effizienzgründen werden anstelle der alpha-numerischen, symbolischen Daten ausschießlich numerisch kodierte Werte benutzt, um die Vergleichsoperationen auf den Datentupeln zu optimieren.

Um das Problem des überdimensionierten Assoziativspeichers beziehungsweise der Gewichtsmatrix $\mathbf{W}$ zu umgehen (Punkt 2 aus Abschnitt 3.3.2), ist es denkbar, sämtliche Tupel $\mathbf{x}$, die zu wenig Ähnlichkeit mit vorhergegangenen, vorher präsentierten, Vektoren aufweisen, abzuspeichern. Diese Liste der Datentupel wird bei jedem Assoziationsvorgang durchlaufen, und die Ähnlichkeit der dort gespeicherten Vektoren mit dem aktuellen verglichen[6]. Konnte keine ausreichende Ähnlichkeit mit den in der Liste enthaltenen Vektoren erzielt werden, so wird die aktuelle Regel als neue Vergleichsregel der bisherigen Liste angehängt und gespeichert. Es handelt sich also um eine Generalisierung der Datentupel, bei der leicht verschiedene Regeln zu einer einzelnen zusammengefaßt werden. Eine solche verallgemeinerte Regel soll mit $\hat{R}_{ij}$ bezeichnet werden.
Auf diese Weise kann das *statische* Modell des Assoziativspeichers in ein *dynamisches* umgewandelt werden, das der Speicherausnutzung in sofern zugute kommt, als daß nur entsprechend einer groben Einteilung der präsentierten Regeln (Muster) Speicherplatz verwendet wird. Das heißt, die Assoziationsschicht wächst nur nach Bedarf mit der notwendigen Anzahl an Regeln. Auf diese Weise wird das Problem des dünn besetzten Assoziativspeichers umgangen.

[5] *engl.*: market-basket analysis

[6] Schon an dieser Stelle käme die numerische Kodierung der einzelnen Werte der Effizienz bei diesen Vergleichsdurchläufen zugute.

3.4.2 Die stochastische Schicht

In der zweiten Schicht sollen die einzelnen Assoziationen, die Ausgaben der ersten, der Assoziationsschicht, bezüglich ihres Wahrscheinlichkeitswertes gewichtet werden. Auf diese Weise sollen Mehrdeutigkeiten, die entstehen, wenn zwei verschiedene Eingaben auf dieselbe Mißbrauchsart abbilden, auf Basis wahrscheinlichkeitsabhängiger Datengrundlagen aufgelöst werden. Dabei soll auf die statistischen Merkmale der Regeln Bezug genommen werden.

Im Bereich des Data-Mining beziehungsweise der Warenkorbanalyse werden Berechnungsmethoden angewandt, um die gefundenen Regeln entsprechend der statistischen Vorkommen einordnen und bewerten zu können. Außerdem sind in diesen Bereichen der Datenverarbeitung Algorithmen entworfen worden, die helfen, schnell und effizient, spezialisierte Regeln zu generieren, die eine vorgegebene Situation auf Basis der zugrundeliegenden Daten noch ausreichend zuverlässig beschreiben. Aus diesem Grund soll im folgenden Abschnitt auf die Methodik und Arbeitsweise des Data-Mining eingegangen werden, sowie Parallelen zum vorliegenden Problem der wahrscheinlichkeitsgestützten Musterassoziation aufgezeigt werden.

3.5 Data-Mining Theorie

Im Prinzip handelt es sich beim Data-Mining um ein Verfahren, das es erlaubt, in extrem großen Datenbeständen, wie sie zum Beispiel bei Supermärkten mit Hilfe der Registrierkassen erstellt werden können, oder in den zunehmenden Data-Warehouse[7]-Umgebungen, nützliche Daten zu extrahieren und neue – bisher unbekannte – Zusammenhänge aufzudecken [Bol96]. Es handelt sich dabei also um eine Art moderner Datenanalyse, bei der der Benutzer selbst möglichst wenige Annahmen oder Hypothesen über die eigentlichen Datenzusammenhänge a priori kennen muß. Zweck einer solchen Analyse sind oft Assoziation, Klassifikation, Segmentierung (*Clustering*) oder Induktion von Regeln aus dem bestehenden Datenbestand. Im Zusammenhang zum Beispiel mit einem *Data-Warehouse* ergibt sich damit ein mächtiges Mittel, Profile und Verhaltensweisen aus dem sorgfältig gepflegten Datenbestand zu erkunden und eventuell sogar unbekannte Korrelationen zwischen zuvor als unwesentlich erachteten Eigenschaften zu finden. Auch in Bezug auf die Mißbrauchsprävention können solche Datenbestände in Zusammenhang mit den Techniken des Data-Mining erfolgsversprechend eingesetzt werden. Der Bericht von Toni Bollinger [Bol96] gibt einen guten Überblick über die Verwendung von Assoziationsregeln in Zusammenhang mit Data-Mining-Verfahren,

[7] Ein *Data-Warehouse* ist eine stark vernetzte Datenbank, in der sämtliche Informationen und Daten eines Unternehmens zusammengefaßt und untereinander so weit wie möglich in Relation gesetzt werden.

und dient als Grundlage der folgenden Erörterungen.

Um Korrelationen zwischen verschiedenen Datensätzen im Datenbestand aufzudecken, werden Regeln gesucht, die mit gerade noch ausreichender Qualität einen potentiellen Zusammenhang auf Basis der Daten beschreiben. Die Qualität drückt sich in den entsprechenden Auftrittswahrscheinlichkeiten aus, mit denen diese Regeln beschrieben werden können.

3.5.1 Begriffserläuterungen

Die Bezeichnung *Regel* oder *Assoziationsregel* stammt selbst aus dem Bereich des Data-Mining und beschreibt eine Verknüpfung zweier Eigenschaften. Eine Regel setzt sich somit aus zwei Teilen zusammen, die die Grundlage für die entsprechenden statistischen Berechnungen sind. Eine Regel besteht aus dem Regelrumpf X und einem Regelkopf Y. X und Y müssen disjunkt sein und bestehen aus einer Menge von *Eigenschaften*[8] oder Elementen, die auch als Entitäten aufgefaßt werden können. Der Regelrumpf wird auch als *Prämisse*, der Regelkopf als *Konklusion* bezeichnet. Eine Regel wird oft wie folgt dargestellt:

$$X \longrightarrow Y \tag{3.13}$$

Diese Assoziationsregeln werden zum Beispiel bei Problemen wie der Warenkorbanalyse eingesetzt, bei der es darum geht, verschiedene Waren aus dem Sortiment untereinander pro Einkaufstransaktion in Beziehung zu setzen – zu assoziieren. Eine solche Transaktion t erfüllt eine Regel $X \rightarrow Y$, wenn $(X \cup Y) \subset t$ gilt, also alle Elemente aus X und Y in der Transaktion vorkommen.
Im Bezug zur vorliegenden Problematik kann somit der Anteil der symbolischen Daten den Regelrumpf und die assoziierte Mißbrauchsklasse den Regelkopf beschreiben. Eine Regel läßt sich dann als

$$x_i \longrightarrow y_j \qquad \text{beziehungsweise als} \qquad x_i \longrightarrow M_j$$

einordnen.

Desweiteren werden folgende Werte benutzt, mit deren Hilfe man Aussagen über die Qualität der Regeln machen kann:

Support: Als den Support bezeichnet man die Anzahl der Regeln, in denen ein zuvor bestimmter Regelrumpf (X) in Verbindung mit einem bestimmten Regelkopf (Y) auftritt im Verhältnis zu der Gesamtanzahl aller Regeln[9]

[8] *engl.*: items
[9] legale sowie illegale Transaktionen = (R_{ij})

($\|\mathcal{D}\|$).

$$support(X \rightarrow Y) = \frac{\|\{t \in \mathcal{D} | (X \cup Y) \subseteq t\}\|}{\|\mathcal{D}\|} \tag{3.14}$$

In Worten der Stochastik ist der Support die Wahrscheinlichkeit, mit der eine Regel in der kompletten Transaktionsmenge $\mathcal{D}$ auftaucht – die relative Häufigkeit. Man schreibt:

$$support(X \rightarrow Y) = P(X \cup Y) = \frac{\|R_{ij}\|}{\|\mathcal{D}\|} \tag{3.15}$$

Unter Umständen ist auch von „prior probability“ die Rede [BG98]. Umgangssprachlich läßt sich der Support beschreiben als:

$$support = \frac{\#\text{ Vorkommen der assoziierten Eigenschaften}}{\#\text{ aller Transaktionen}}$$

Als Beispiel in Bezug zu den symbolischen Transaktionsdaten diene eine Regel, die besagt, daß mit einer hohen Wahrscheinlichkeit – dem Support – Transaktionen in D-Mark mit zum Beispiel der Eurocard[10] getätigt werden. Als Beispiel für einen niedrigen Support sei eine sehr hoch spezialisierte Regel angegeben, die eine sehr ungewöhnliche Kombination aus Eigenschaften enthält. Diese Regel ist für die weitere Verarbeitung nicht von Bedeutung, da sie eine Ausnahme darstellt und nicht weiter repräsentativ eingesetzt werden kann. Eine solche Regel braucht weiterhin also nicht mehr betrachtet werden.

Konfidenz: Unter Konfidenz[11] versteht man das Verhältnis aus dem Support der kompletten Regel zum Support des Regelrumpfes. Es handelt sich also in Worten der Stochastik um die **bedingte Wahrscheinlichkeit** (abhängige Wahrscheinlichkeit) des Regelrumpfes. Es gilt:

$$confidence(X \rightarrow Y) = P(Y|X) = \frac{P(X \cup Y)}{P(X)} \tag{3.16}$$

oder anders

$$confidence(X \rightarrow Y) = \frac{\|\{t \in \mathcal{D} | (X \cup Y) \subseteq t\}\|}{\|\{t \in \mathcal{D} | X \subseteq t\}\|} = \frac{support(X \rightarrow Y)}{support(X)} \tag{3.17}$$

Das heißt, die Konfidenz bestimmt den Anteil eines Elementes X in der kompletten Regelmenge in Abhängigkeit von Y. Hier ist von „conditional

[10] Dies dürfte in sehr vielen Fällen zutreffen

[11] confidence := (*engl.*) Vertrauen

probability“ die Rede [BG98]. Die Konfidenz bezüglich der Regeln kann also auch bezeichnet werden als

$$confidence = \frac{\#\text{ abgedeckte Mißbrauchstransaktionen}}{\#\text{ aller abgedeckter Transaktionen}}$$

Als Beispiel diene folgender konstruierter Fall: Eine Regel deckt Mißbrauchstransaktionen ab, die über das Internet getätigt wurden. Diese Regel hat eine hohe Konfidenz, da für Autorisierungsanfragen über das Internet eine hohe Mißbrauchsquote existiert. Das heißt, bei vielen der Kreditanfragen auf diesem Wege handelt es sich tatsächlich um versuchte Mißbräuche; selten kommt es zu einem Fehlalarm aufgrund dieser Regel. Desweiteren werden im Verhältnis bisher eher wenige Transaktionen über dieses Medium getätigt, so daß der Nenner in etwa dieselbe Mächtigkeit erlangt wie der Zähler in Gleichung 3.17. Anders verhält es sich zum Beispiel bei stark verallgemeinerten Regeln, die sehr viele legale und auch illegale Transaktionen vertreten. Die Konfidenz fällt dabei klein aus, da durch die überwiegend legalen Transaktionen der Nenner überproportional zunimmt. Mit einer solchen Regel ist eine Mißbrauchszuordnung nicht ausreichend zuverlässig möglich.

Abdeckung: Die Abdeckung[12] gibt an, mit welcher Wahrscheinlichkeit eine Regel im speziellen Datenbereich – im vorliegenden Fall den Mißbrauchsdaten – auftritt [HH98]. Dies kann als ein Maß der Repräsentanz einer Regel in dem jeweiligen, speziellen Datenbereich angesehen werden. Die Abdeckung berechnet sich wie folgt, wobei $\mathcal{M}$ in diesem Fall die Menge der Transaktionen des jeweiligen Datenbereichs ist:

$$share(X \rightarrow Y) = \frac{\|\{t \in \mathcal{M} | (X \cup Y) \subseteq t\}\|}{\|\mathcal{M}\|} \tag{3.18}$$

Die Abdeckung ist ein entscheidender Faktor bei der Einstufung der verallgemeinerten Mißbrauchsregel. Stark spezialisierte oder einmalig auftretende Mißbrauchsregeln können nur wenige Mißbrauchstransaktionen aufdecken. Auf diese Weise tragen sie nicht zur Reduzierung des Mißbrauchsdatenraumes durch Abstraktion bei. Aus diesem Grund sollen möglichst universelle Mißbrauchsregeln generiert, beziehungsweise verwendet werden, um möglichst viele Mißbräuche – nicht aber legale Transaktionen – abzudecken. Die Abdeckung soll also einen maximalen Abdeckungswert erreichen, oder zumindest eine vorgegebene Abdeckungsschwelle überschreiten. Anders als der Support bezieht sich die Abdeckung also nur auf eine spezielle Transaktionsklasse, entweder auf die legalen Transaktionen oder auf die illegalen. Meist ist jedoch im Zusammenhang mit der Mißbrauchsanalyse die Rede

[12] *engl.*: share

von der Abdeckung bezüglich der Mißbrauchsdatensätze, wenn nicht anders vermerkt. Die Abdeckung kann also in diesem Zusammenhang als

$$share = \frac{\#\text{ abgedeckte Mißbrauchstransaktionen}}{\#\text{ alle Mißbrauchstransaktionen}}$$

angegeben werden.
Als Beispiel diene erneut eine Regel, die Transaktionen, die über das Internet getätigt wurden, vertritt. Hier ist die Abdeckung bezüglich des Mißbrauchs sehr hoch, es treten verhältnismäßig viele Mißbrauchstransaktionen mit dieser Eigenschaft in diesem Datenraum auf. Hingegen ist die Abdeckung bezüglich der legalen Transaktionen nur gering, da der kleine Anteil an legalen Internettransaktionen in der großen Menge dieser Transaktionen untergeht.

3.5.2 Regelfindung

Nachdem nun die wesentlichen Begriffe eingeführt sind, soll nun ein Regelfindungsprozeß nach [AS94] beschrieben werden, um die Vorgehensweise beim Data-Mining zu verdeutlichen. Um die Assoziationsregeln für zum Beispiel eine Warenkorbanalyse zu finden, geht man von den einzelnen Elementen oder Merkmalen aus, sogenannten *Items* und versucht immer größere Regeln aus diesen Items zu bilden, die die Minimalanforderungen in Form einer Mindestkonfidenz und des Mindestsupports des Regelfindungsprozesses erfüllen. Dabei werden *die* Regeln als *häufig* bezeichnet, deren Support den zuvor festgelegten Mindestsupport übertrifft.

Der sogenannte `Apriori-Algorithmus` aus [AS94] besteht aus 2 Phasen:

1. Bestimmung der häufigen Regeln auf Basis des Supports für die aktuelle Regelgröße durch Zählläufe auf der Datenbasis.

2. Anhand des Supports kann dann die Konfidenz entsprechend Gleichung 3.17 berechnet werden. Die Regeln, die der Mindestkonfidenz genügen, bleiben erhalten und werden erneut mit allen noch nicht beteiligten Items kombiniert.
   Es wird mit Schritt 1 fortgefahren.

Bei der Berechnung der häufigen Itemmengen macht man sich hier zunutze, daß gilt: Wenn X eine häufige Itemmenge ist, dann ist auch $X' \subset X$ häufig! Damit brauchen im $n+1$ Iterationsschritt jeweils nur noch die Datensätze durchsucht werden, die die Teilitemmenge X' schon enthalten, alle übrigen werden nicht die Bedingung des Mindestsupports erfüllen.

Es ist festzuhalten, daß der beschriebene Algorithmus anfängt, die Regeln aus den einzelnen Elementen zu bilden und diese zu immer spezielleren Regeln mit anderen Elementen erweitert. Um diese Spezialisierung in Bezug auf die Elemente zu verallgemeinern, wurde der Algorithmus in [SA95] dahingehend erweitert, daß Hierarchieklassen – benutzerdefinierte, generalisierte Oberklassen – in den zugrundeliegenden Regeln zugelassen sind.

3.6 Die stochastische Schicht in Bezug zum Data-Mining

Um nun bei mehrdeutigen Mißbrauchsassoziationen eine wahrscheinlichkeitsabhängige Entscheidung bezüglich der Mißbrauchsarten zu treffen, muß nun ein Maß gefunden werden, das als Gewicht p_{ij} in dem geplanten Netzmodell dienen kann. Im folgenden wird dazu der Bezug zur Anwendung und Terminologie des Data-Mining hergestellt.

Sei r_{ij} die Anzahl der Regeln, die von der Regel R_{ij} abgedeckt werden, zunächst also $r_{ij} = \|R_{ij}\|$. Mit r sei die Menge aller Regeln bezeichnet, also $r = \sum_i \sum_j \|R_{ij}\|$. Entsprechend sei $\hat{r}_{ij}$ die Anzahl der von einer verallgemeinerten Regel abgedeckten Transaktionsdatensätze. Damit ergibt sich dann entsprechend Gleichung 3.14:

$$support(\mathbf{x}_i \rightarrow (M_j = 1)) = support(R_{ij}) = \frac{r_{ij}}{r} \tag{3.19}$$

Demzufolge läßt sich die bedingte Wahrscheinlichkeit p_{ij}, die Konfidenz der Regel $R_{ij} = \mathbf{x}_i \rightarrow M_j$ entsprechend Gleichung 3.17 wie folgt bestimmen:

$$p_{ij} \equiv confidence(\mathbf{x}_i \rightarrow M_j) = confidence(R_{ij}) = \frac{support(r_{ij})}{support(\mathbf{x}_i)} \tag{3.20}$$

wobei für den Support von $\mathbf{x}_i$, den Regelrumpf, gilt:

$$support(\mathbf{x}_i) = \sum_{j=0}^{p} support(R_{ij}) \tag{3.21}$$

Die bedingte Wahrscheinlichkeit p_{ij} für das Auftreten des Mißbrauchs M_j beim Vorliegen eines Eingabetupels $\mathbf{x}_i$ kann nun als Gewicht für die Bewertung der Ausgabe A_j der Assoziativschicht auf Eingabe $\mathbf{x}_i$ gewählt werden. Die Assoziation A_j wird somit entsprechend ihrer Auftrittswahrscheinlichkeit gewichtet. Im Falle einer Mehrdeutigkeit kann bei der Mißbrauchszuweisung für die „wahrscheinli-

chere“ Mißbrauchsart beziehungsweise Assoziation entschieden werden:

$$\begin{aligned} M_k(\mathbf{x}) = S_k^M(A_j(\mathbf{x})) &\equiv confidence(\mathbf{x} \rightarrow M_j) \\ &= \sum_i confidence(\mathbf{x}_i \rightarrow M_j) \cdot support(\mathbf{x}_i) \\ &= \sum_i p_{ij} A_j(\mathbf{x}_i) \end{aligned} \tag{3.22}$$

$M_k(x)$ ist damit die Ausgabe der 2. Schicht, durch die in Gleichung 3.22 abgebildete Berechnungen des k-ten Neurons in dieser Schicht gegeben. Die Auftrittswahrscheinlichkeit in Form des Supports des Eingabetupels $\mathbf{x}_i$ muß zunächst in Gleichung 3.22 berücksichtigt werden, um Gleichheit zu gewährleisten, kann letztendlich jedoch außer acht gelassen werden, da nun ausschließlich die Konfidenz p_{ij} als Gewichtung bezüglich der Assoziationen $A_j(x_i)$ verwendet wird.
Im Zuge des Trainings werden sich die Zählerstände beziehungsweise die Konfidenzwerte zunehmend stabilisieren. Mit dieser Behandlung der Gewichte in der stochastischen Schicht erübrigt sich außerdem ein lernratenabhängiges Training. Eher bietet es sich an, die absoluten Zählerstände r_{ij} zur Berechnung der Konfidenz zu verwenden und diese mit den einzelnen Regeln mitzuführen.

3.7 Ausgabeschicht

In der abschließenden Ausgabeschicht werden nun die Ergebnisse der einzelnen Mißbrauchsarten mit Ausnahme der legalen Transaktionen zusammenaddiert. Hier genügt ein lineares Neuron, das die Summe der wahrscheinlichkeitsgestützten Mißbrauchsassoziationen $M_k(\mathbf{x})$ mit $k = \{1, \ldots, p\}$ bestimmt und einer Schwelle s_m unterzieht. Überschreitet diese Summe den zuvor festgelegten Wert s_m, so kann mit ausreichender Sicherheit davon ausgegangen werden, daß ein tatsächlicher Mißbrauch vorliegt.

$$M(\mathbf{x}) = \sum_{k=1}^{p} M_k(\mathbf{x}) \tag{3.23}$$

Nach Aussage der GZS ist eine Schwelle von 10% als Richtmaß für die Auftrittswahrscheinlichkeit einer Regel zu wählen, um diese ausreichend als Mißbrauchsregel einzustufen. Kann eine Regel mit einer Konfidenz p_{ij} von mehr als 10% mit einem Mißbrauch M_j mit $j > 0$ assoziiert werden, so kann der entsprechende Mißbrauch akzeptiert werden. Man nennt dies auch das Bayes-Kriterium einer Klassenentscheidung [BG98]. Mit dieser wahrscheinlichkeitsgestützten Klassifizierung sind die Nachteile 1 und 4 des einfachen Assoziativspeichers aus Abschnitt 3.3.2 beseitigt.

Wie aus Gleichung 3.23 ersichtlich, soll prinzipiell von nun ab *nur* noch nach Mißbrauch oder nicht Mißbrauch entschieden werden. Damit erübrigt sich die Mitführung der Mißbrauchsart im Regelkopf; es reicht nun, für die notwendigen Berechnungen den Regelrumpf im jeweils zu untersuchenden Datenbestand auszuwerten. Dies entspricht einer binären Entscheidung, für die gilt:

$$M(\mathbf{x}) = \begin{cases} -1 & \text{Mibrauch} \\ +1 & \text{kein Mißbrauch} \end{cases} \qquad \begin{matrix} \sum_{k=1}^{p} M_k(\mathbf{x}) \geq 0,1 \\ sonst \end{matrix} \tag{3.24}$$

3.8 Optimierung des statistischen Assoziativspeichers

Durch die Hinzunahme neuer, von den bisherigen abweichenden Regeln, konnte das Modell des Assoziativspeichers in ein dynamisches, sich anpassendes Netz umgewandelt werden. Nichtdestotrotz ist das Verlernen unnötiger Regeln beziehungsweise das Löschen überflüssiger Regeln noch nicht vorgesehen. Die Größe der Mustermenge kann somit unter Umständen bis auf den „worst case“, die Größe aller möglichen Permutationen, anwachsen. Einmalig präsentierte, jedoch nicht repräsentative Muster sollen aus diesem Grund möglichst erkannt und nach einiger Zeit beziehungsweise einigen „Lernschritten“ wieder eliminiert werden.

Ein solches „Verlernen“ kann mit Hilfe der Konfidenzgewichte aus der 2. Schicht umgesetzt werden. Dabei werden die Gewichte p_{ij} beobachtet. Fällt die berechnete Konfidenz einer Regel unter eine festgesetzte Schranke, also tritt die Regel R_{ij} nicht genügend oft als Kombination aus Eingabevektor $\mathbf{x}_i$ und Mißbrauch M_j auf, so kann dieses Gewicht annulliert werden. Gilt dies für sämtliche Mißbrauchsarten, die mit dem Regelrumpf assoziiert sind, so können die Regeln R_{ij} mit $j = \{1, \ldots, p\}$ und damit die entsprechende Assoziation $A_j(\mathbf{x}_i)$ samt der Gewichte p_{ij} gelöscht werden. Das heißt, der Eingabevektor $\mathbf{x}_i$, bestehend aus den numerisch kodierten Werten kann aus der Liste der gespeicherten Mustervektoren entfernt werden.
Ein solches Verlernen kann auch provoziert werden. Vorstellbar in diesem Zusammenhang ist eine Division der einzelnen Zählerstände r_{ij} beziehungsweise $\hat{r_{ij}}$ durch zwei, die mit einer Registerverschiebung schnell und effizient durchgeführt werden kann. Dadurch wird die Konfidenz nicht verändert, da sowohl Zähler als auch Nenner (vergleiche Gleichung 3.17) von dieser Berechnung gleichermaßen betroffen sind. Stattdessen kommt es zu Auswirkungen auf Support und Abdeckung. Trotzdem führt dieses Verfahren fortgesetzt dazu, daß Regeln, die nur sehr selten auftreten, letztendlich irgendwann mit $r_{ij} = 1$ vertreten sind, was bei der nächsten Division aufgrund der ganzzahligen Werte zu $r_{ij} = 0$ führt und damit zu einer Konfidenz von $c = 0$. Damit ist die festgesetzte Konfidenzschwelle auf jeden Fall unterschritten, und es kann, wie oben beschrieben, weiterverfahren

werden. Problematisch gestaltet sich dieses Verfahren nur im Zusammenhang mit neu angefügten Mißbrauchsregeln, deren Zählerstände aus diesem Grunde noch klein sind.
Auf diese Weise kann das Modell quasi als „Online"- Lernverfahren angesehen werden, da es sich ständig an die aktuellen, statistischen Gegebenheiten der Eingabedaten anpaßt und so jeweils den momentanen Zustand der Assoziationsbeziehungen beurteilen kann. Damit gilt nun auch Problempunkt 3 aus Abschnitt 3.3.2 als behoben.

3.9 Die Generalisierung

3.9.1 Motivation

Das dynamische Hinzufügen und Entfernen der Regelmuster im Falle des Assoziativspeichers kann durch eine konsequente Generalisierung *aller* Mißbrauchsregeln auf einer statistischen Grundlage explizit ausgeführt werden. Auf diese Weise kann von Anfang an eine genaue Vorgabe gemacht werden, welchen Umfang der Eingaberaum umfassen soll, sowie welchen statistischen Voraussetzungen die einzelnen Regeln genügen sollen. Es bietet sich also an, die 1. und 2. Schicht zu einer einzelnen zusammenzufassen, bei der Assoziation und wahrscheinlichkeitsgestützte Gewichtung auf Basis eines Data-Mining-Verfahrens wie in 3.5.2 beschrieben, ausgeführt werden. Aus diesem Umstand heraus wurde versucht, einen Algorithmus zu entwickeln, der mit der nötigen Genauigkeit und Präzision die vorliegenden Mißbrauchsregeln, bestehend aus den symbolischen Werten der Kreditkartentransaktionen und der Konteninhaberdaten, verallgemeinert und statistisch auswertet. Diese Generalisierung soll dabei in der Art stattfinden, daß die festgelegte Mindestkonfidenz für die einzelnen Regeln *nicht* unterschritten wird.
Dieses Vorgehen unterscheidet sich dahingehend von den allgemein üblichen Data-Mining-Ansätzen der Warenkorbanalyse, indem in diesem Fall von dem *vollständigen* Regelrumpf ausgegangen und versucht wird, diesen Schritt für Schritt zu verallgemeinern, wohingegen bei dem in Abschnitt 3.5.2 beschriebenen Verfahren von [AS94] versucht wird, von den einzelnen Werten ausgehend, eine möglichst aussagekräftige Regel mit so viel Elementen wie möglich zu erhalten. Letzteres ist in dem Zusammenhang der Reduzierung des Regelraumes wenig sinnvoll, da es zu Wertekombinationen kommen kann, die unter Umständen gar nicht in Verbindung mit Mißbrauch, wenn überhaupt, vorkommen. Auf diese Weise erhielte man einen unnötig erweiterten Eingaberaum an potentiellen, aber dennoch in diesem Zusammenhang womöglich unrealistischen Regeln. Aus diesem Grund soll die nun im folgenden beschriebene Generalisierung zum Einsatz kommen. Dabei wird von den Regeln auf Basis der realen Transaktionsdaten ausgegangen. Diese Regeln werden dann schrittweise verallgemeinert. Die durch die Generalisie-

rung erzeugten, einfachen, unspezifischen, aber dennoch prägnanten Regeln sollen dazu verwendet werden, um mit Hilfe nur weniger Elemente, schnell dem symbolischen Anteil der eintreffenden Transaktionen die entsprechende Mißbrauchsart zuzuordnen. Es liegt eine stark verminderte, aber dennoch statistisch abgesichterte Regelmenge nach solch einem Generalisierungsschritt vor.

3.9.2 Ein Modell zur Generalisierung

Es wird nach unspezifischen Mißbrauchsregeln gesucht, die mit möglichst wenigen, eindeutig beschriebenen Elementen beziehungsweise Items auskommen und trotzdem eine ausreichend hohe Konfidenz haben. Darüber hinaus sollten diese verallgemeinerten Regelrümpfe ein festgelegtes Minimum der Abdeckung auf den Mißbrauchsdaten erreichen, um als ausreichend repräsentativ zu gelten.
Mit der angestrebten Generalisierung soll der Eingaberaum der Assoziativschicht weiter eingeschränkt werden. Die übrigbleibenden Prototypen der ursprünglichen Mißbrauchsregeln enthalten damit nur noch die wesentlichen, mißbrauchsspezifischen Eigenschaften in den Elementen. Die übrigen Elemente (Items) werden durch so genannte *Wildcards*, mehr oder weniger universelle Platzhalter, ersetzt. Diese Wildcards werden im Weiteren auch mit „$\star$“ bezeichnet.

Der Aufbau der Generalisierung gestaltet sich im wesentlichen folgendermaßen:

1. Distanzbestimmung, Vergleich der Regeln durch Abstandsbestimmung

2. Regelzusammenführung, Verallgemeinerung durch Einfügen von Wildcards an Stellen mit einem Unterschied

3. Bestimmung der jeweiligen Werte für Konfidenz, Abdeckung und eventuell Support

Zu Beginn liegen sämtliche bekannte mit Mißbrauch assoziierte Prämissen, den symbolischen Datentupeln der Mißbrauchstransaktionen, in einer Arbeitsliste vor. Es kann durchaus zu diesem Zeitpunkt zu doppelt auftretenden solcher Mißbrauchsregeln kommen, da es sich um die von den Analogdaten unabhängigen Datensätze handelt und somit eindeutige Kriterien wie Transaktionsdatum und -zeit für die verschiedenen Transaktionen wegfallen. Diese Arbeitsliste wird nun mit jedem weiteren Generalisierungsschritt mehr und mehr durch Zusammenlegung verschiedener Regelpaare dezimiert, sofern es die Konfidenz und Abdeckungswerte zulassen.
Als Abbruchbedingung der Generalisierung gilt,...

1. ... wenn die generalisierten Mißbrauchsregeln ausschließlich aus Wildcards bestehen.

2. ... wenn keine Regel mehr in der Liste der potentiell zu generalisierenden Regeln enthalten ist.

3. ... wenn eine vorgegebene Iterationsgrenze erreicht ist.

Sind alle Regeln am Ende eines Generalisierungslaufes miteinander verglichen, werden die zur Generalisierung benutzen Regeln entfernt, und die neu gebildeten Regeln werden an die Arbeitsliste angehängt, so daß erneut wieder alle ursprünglichen Mißbrauchsregeln von den Regeln dieser Liste abgedeckt werden. Diese Vorgehensweise erinnert auch an das von [Toi] beschriebene Verfahren zur Regelgruppierung und -generalisierung. Diese Liste gilt als Grundlage für einen neuen Generalisierungsdurchlauf und dient als Regelbasis für die anschließende Mißbrauchsdiagnose.
Im Unterschied zu den allgemeinen Data-Mining-Verfahren wie sie in [AS94] beschrieben werden, ist jedoch eine langwierige Generalisierung notwendig, um auf tatsächlich kurze, auf wenige Merkmale spezialisierte Regeln zu gelangen. Hier ist ein Vorteil in den Verfahren zu sehen, die versuchen, die Regeln aus den einzelnen Datenelementen aufzubauen. Es ist damit in kurzer Zeit möglich, Regeln aus nur wenigen Elementen unter den vorgegebenen Bedingungen zu generieren. Jedoch kann es sich bei den auf diese Weise generierten Regeln um unrealistische Kombinationen aus den symbolischen Daten handeln, die für die vorliegende Problembetrachtung nur hinderlich und nicht von Bedeutung sind. Darum überwiegt der Vorteil bei der Generalisierung, ausschließlich nur auf den tatsächlich vorkommenden Daten beziehungsweise Datenkombinationen, zu arbeiten und damit den Datenraum so klein wie möglich halten zu können.

Im folgenden sollen nun die einzelnen Schritte, wie sie unter anderem auch im Flußdiagramm 3.5 aufgeführt sind, eingehender erläutert und an Beispielen verdeutlicht werden.

Die Entropie entsprechend Abschnitt 3.2 eines Datums liefert zusätzliche Informationen, die zur Laufzeitersparnis und zur Genauigkeit bei der Generalisierung beitragen. Die Entropie ist ein Maß der Information, das hypothetisch aufgestellt wird. Erreicht die Entropie ein Maximum beziehungsweise einen hohen Wert, so ist von einer eher gleichverteilten, stark gestreuten Datenverteilung des Merkmals auszugehen. Im Gegenteil, geht die Entropie gegen 0, so kann von einer kleinen Streuung der Daten ausgegangen werden. Die Entropie gibt also eine Wertung bezüglich der Spezialisierung der symbolischen Daten eines Datentyps an.
Diese Eigenart der Entropie wird nun bei der Bewertung der Wildcards sowie bei der Sortierung der Vergleichsreihenfolge der einzelnen Daten benutzt. Wie in Tabelle E.1 in Anhang E aufgeführt, wurden sämtliche Entropiewerte für die vorliegenden symbolischen Daten jeweils zusammen und für die einzelnen Datenbereiche getrennt entsprechend Gleichung 3.6 auf Seite 22 berechnet.

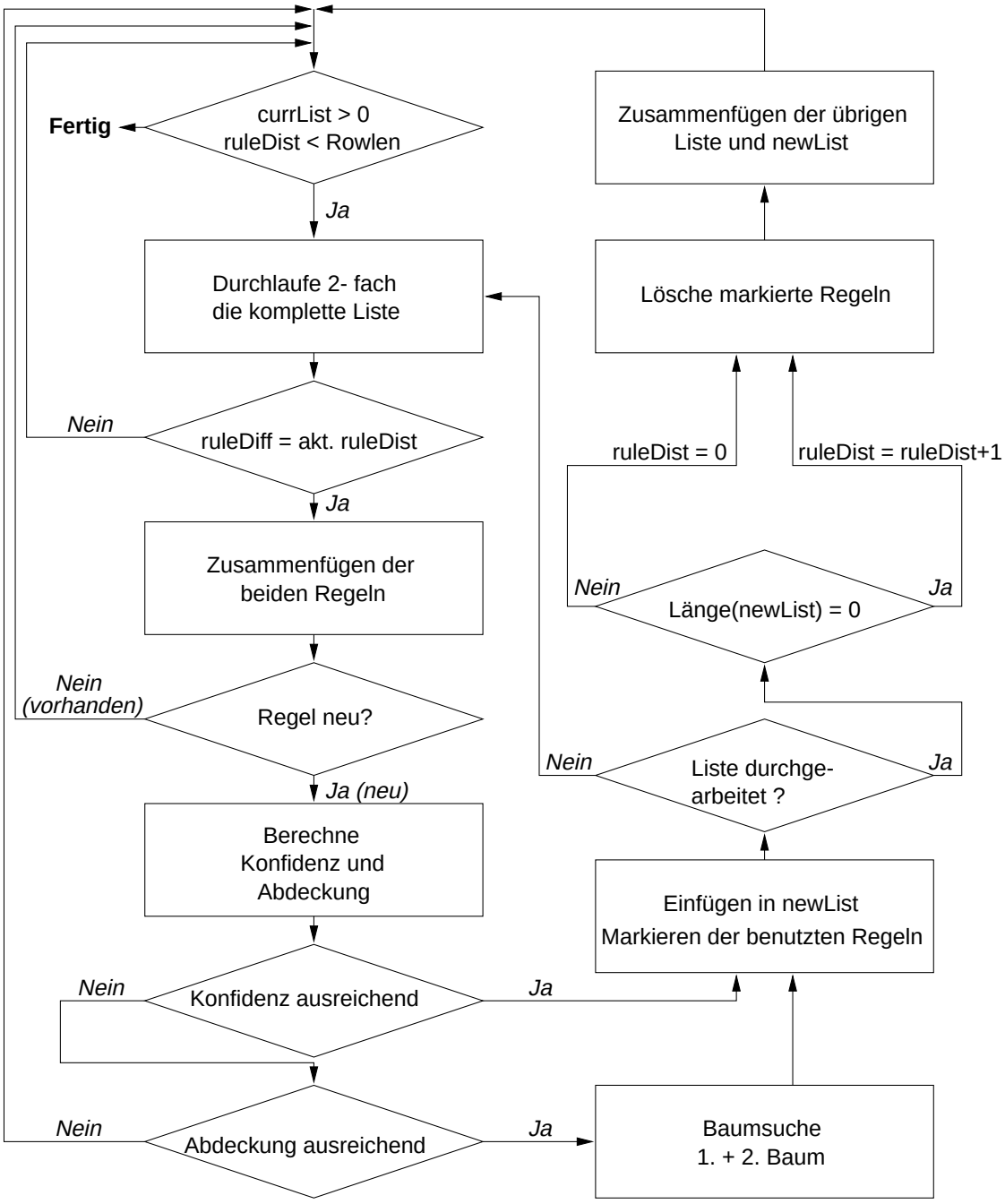


Abbildung 3.5: Flußdiagramm der Generalisierung

Bestimmung der Data-Mining-Werte

Es ist angebracht, die Konfidenzbestimmung nur auf einem *Sample*[13] der legalen Daten durchzuführen, da sonst die Menge dieser Datensätze die Laufzeit der Berechnungsroutine sprengt. Dieses Sample ist zufällig aus den legalen Datensätzen zu bestimmen. Mit der Größe dieses Datenauszuges steigt die Genauigkeit der Berechnung der Wahrscheinlichkeitswerte. Auch ist es von grundlegender Bedeutung für die Konfidenzberechnung, repräsentative Datensätze für diesen Datensample zu wählen, da in Folge von Hochrechnungen auf die reale Mächtigkeit der legalen Daten im Verhältnis zu den illegalen Daten leicht Abweichungen um den Faktor 100 und mehr entstehen können.
Für das Ermitteln der Absolutzahlen werden also jedesmal die gesamten Teilmengen (Sample) an legalen und illegalen Datensätzen durchsucht und verglichen. Bei diesem Schritt rentiert sich eine Optimierung durch numerisch kodierte Werte, die der Speicherstruktur des benutzten Rechners entspricht, um damit die Vergleichsroutinen zu beschleunigen. Aufgrund der Kodierung in Zahlenwerte kann ein solcher aus nur 26 Zahlen bestehender Regelrumpf mit Hilfe systemeigener, optimierter Vergleichsroutinen schnell mit allen ebenso gespeicherten Mißbrauchsregeln und legalen Regeln verglichen werden.

Die resultierenden Absolutzahlen müssen jedoch, um dem tatsächlichen Verhältnis aus legalen und illegalen Datensätzen gerecht zu werden, im Falle der legalen Datensätze auf ein reales Maß hochgerechnet werden. Dazu wird als Hochrechnungsfaktor h das Verhältnis aus der hochgerechneten Anzahl von legalen Datensätzen und der Samplegröße der legalen Daten verwendet:

$$h := \frac{\#\text{Mißbrauchsdatensätze} \cdot 1000}{\text{Samplegröße}} = \frac{5850000}{30000} = 195 \qquad (3.25)$$

Im Nachhinein können die Ergebnisse zu Testzwecken auf den kompletten Daten zusätzlich getestet und verifiziert werden.

Das eigentliche Modell des Assoziativspeichers kann im Zuge der Realisierung nur noch als Konzept dienen, wobei die einzelnen Schritte durch optimierte Operationen mit Mitteln der konventionellen Datenverarbeitung erzielt werden. So werden die Mißbrauchsassoziationen nunmehr durch die generalisierten Regeln selbst repräsentiert. Eine Verrechnung auf Basis des dünn besetzten Assoziativspeichers mit Hilfe boolschen Operationen wäre in diesem Zusammenhang, selbst auf Rechenmaschinen mit großen Registergrößen, unrentabel, da auch in diesem Fall die Rechenoperationen auf dem mehrere tausend Bit breiten Eingabefeld sequentiell abgearbeitet werden müßten.

[13] eine Untermenge der jeweiligen Datensätze

Die Abstandsberechnung

Um die Eingabevektoren auf ihre Ähnlichkeit hin zu überprüfen ist es notwendig, ein Abstandsmaß zu bestimmen. Eine Abstandsberechnung, wie sie bei analogen Daten mit einer „Nachbarschaftsbeziehung“ vorliegt, ist bei dieser Art von Daten nicht sinnvoll und nicht möglich. Somit wurde der Abstand zweier Regeln definiert als:

Definition 1 *Der Abstand ist die Anzahl der sich unterscheidenenden Regelwerte mit einer Toleranz ϵ.*

Somit sind zwei Regeln gleich, stimmen sie in allen Werten überein. Eine Ähnlichkeit zweier Regeln ist festzustellen, wenn die Anzahl der sich unterscheidenden Werte eine festgelegte Schranke nicht überschreitet.
Sobald die Schranke in einem Vergleichsdurchlauf bezüglich einer Regel überschritten wird, kann an dieser Stelle die Vergleichsroutine frühzeitig abgebrochen werden, was die einzelnen Vergleichsroutinen effizienter gestaltet.
Um den eigentlichen Vergleich, den Lauf über alle Datenwerte der aktuell zu vergleichenden Regeln, zu beschleunigen und zu optimieren, wurde die Vergleichsreihenfolge derart gewählt, daß zuerst die Datentypen verglichen werden, die die höchste Mißbrauchsentropie aufweisen. Bei diesen Daten liegen aufgrund der Entropiewerte stark gestreute Werte vor, so daß die Wahrscheinlichkeit einer Differenz höher ist als bei Daten mit einer kleinen Entropie. Auf diese Weise wird das Auffinden von Unterschieden in den ersten Vergleichsschritten geradezu provoziert, und die einzelnen Vergleichsroutinen können frühzeitiger abgebrochen werden.
Desweiteren sind die Regeln in aufsteigender Reihenfolge zunächst nach der Konfidenz und an zweiter Stelle nach der Abdeckung sortiert, so daß die am weitesten generalisierten Regeln beziehungsweise die Regeln, mit denen eine weitergehende Generalisierung am ehesten zu den gesuchten, allgemeinen Regeln führt, zu Beginn verglichen werden. Weiter kommt es in diesen Fällen am ehesten zu einer Unterschreitung der jeweiligen Konfidenzschwelle. Auf diese Weise werden mit Hilfe dieser Regeln entweder durch weitere Verallgemeinerungen Regeln generiert, die nicht mehr dem Mindestkonfidenzwert entsprechen – der Regelraum verkleinert sich –, oder es sind die bestmöglichen Bedingungen gegeben, eine weitere, mindestens ebenso allgemeine, neue Regel zu generieren.
In den symbolischen Karteninhaberstammdaten sind auch zwei Felder enthalten, die mehr oder weniger rückwirkend einen Mißbrauch implizieren. Zum einen ist es das Feld `ACTV_STAT`, das anzeigt, ob die Karte *aktiv* oder *inaktiv* ist. Weiter ist es das Feld `ACCT_STAT`, in dem verschlüsselt der Sperrgrund, sofern es gefüllt ist, verzeichnet ist. Es macht also keinen Sinn, diese Daten bei der Generalisierung beziehungsweise bei den Vergleichsschritten zu betrachten. Aus diesem Grund werden in den jeweiligen Vergleichsroutinen diese Datenfelder übersprungen. Der

Vollständigkeit halber sind sie aber dennoch in die Regeln mit hinein übersetzt worden und werden unter Umständen sogar auch generalisiert.

Der Generalisierungsschritt

Sind zwei Regeln gefunden, die dem aktuellen Ähnlichkeitsmaß α entsprechen – die Regeln unterscheiden sich also in genau α Werten – so werden diese zu einer neuen zusammengefügt. α wird in diesem Zusammenhang auch als *Distanz* bezeichnet. Dabei wird an den α Stellen der Unterscheidungen jeweils ein neuer Wildcard eingefügt, der die jeweiligen Werte abhängig von der vorliegenden Entropie des Datums vertritt. Die übrigen übereinstimmenden Werte werden beibehalten. Das Ähnlichkeitsmaß α wird beginnend mit 0 nur dann um Eins inkrementiert, wenn bei einem einzelnen Generalisierungslauf über alle Regeln *keine* neue, verallgemeinerte Regel gefunden wurde. Sobald jedoch erneut Regeln aufgrund eines höheren Distanzmaßes gefunden werden können, wird dieses im nächsten Schritt wieder auf den Wert 0 initialisiert.
Eine Regel behält Verweise auf die ursprünglichen Regeln, die als Grundlage für die Verallgemeinerung gedient haben, um den Generalisierungsschritt zurückverfolgen zu können. Es entstehen im Laufe der Generalisierung verschiedene „gerichtete Generalisierungsgraphen“ aus Regeln. Ausgehend von den nicht zur Generalisierung benutzen Regeln können Subgraphen, sogenannte Generalisierungsbäume, in dem Gesamtgraphen gebildet werden. Die Baumwurzeln entsprechen den jeweils am weitesten generalisierten Regeln, die noch nicht im Rahmen einer weiteren Generalisierung benutzt wurden. Die Blätter des Baumes entsprechen den eigentlichen Mißbrauchsregeln auf Basis der Mißbrauchstransaktionsdaten, die in keiner Weise generalisiert sind, also keine Wildcards enthalten. Ein solcher Baum ist in Abbildung 3.6 auf Seite 47 dargestellt. Da sich die Wurzelknoten der einzelnen Subgraphen auf verschiedenen Ebenen befinden können spricht man auch von *Generalisierungsleveln*, in denen sich die unbenutzen Regeln befinden. Die einzelnen Generalisierungslevel enthalten also ausschließlich die bisher zur Generalisierung unbenutzen Regeln, mit der dem Level entsprechenden Anzahl an Wildcards.

Die Spezifität der Wildcards hängt von den Entropiewerten der illegalen und legalen Daten ab. Betrachtet man diese getrennt nach Datenherkunft (s. Tabelle E.1 in Anhang E), so können vier verschiedene Ausprägungen unterschieden werden:

1. Die Entropiewerte **beider** Datenklassen, der illegalen und legalen Daten, sind relativ **hoch**. Auf beiden Seiten wird eine starke Streuung der Daten erwartet. Dies führt dazu, daß dieses Merkmal nur wenig Aussagekraft im Rahmen der Mißbrauchseinstufung besitzt.
   Als Beispiel seien die Daten `CTY_2` oder `AID_CD` genannt.

2. Die Entropiewerte **beider** Datenklassen sind relativ **niedrig**, die vorhandenen Werte scheinen sich auf eine kleine Auswahl zu beschränken. Es kann jedoch keine verlässliche Aussage getroffen werden, in wie weit sich die Auftrittswahrscheinlichkeiten der einzelnen Datenwerte untereinander und zwischen den Klassen unterscheiden. Eine Interpretation der *totalen* Entropie kann hier eventuell jedoch mehr Aufschluß liefern. Trotzdem gilt auch hier, daß auf Basis der Entropie alleine kein Nutzen für die Mißbrauchseinstufung gezogen werden kann.
   Als Beispiel seien hier die Datenfelder `ACT_CD` oder `CNTY_CD_1` genannt.

3. Unterscheiden sich die Entropiewerte, so kann auf jeden Fall auf verschiedene Charakteristika der Datenfelder einzelner Klassen geschlossen werden. Ist die Entropie eines legalen Feldes **hoch**, die des illegalen **niedrig**, so kann auf eine Spezifität der illegalen Werte geschlossen werden. Nur wenige Werte dieses Datums sind mitverantwortlich für die Mißbrauchseigenschaft der Regel.
   Als gutes Beispiel diene hier das Feld `CURR_CD`.

4. Gleiches gilt für den umgekehrten Fall, wenn die illegale Entropie eines Feldes besonders **hoch**, die des legalen Feldes **niedrig** ist. Hier kann das Fehlen ganz spezieller Eigenschaften, die in der legalen Klasse in diesem Datum vertreten sind, eventuell als Indiz für einen Mißbrauch gedeutet werden.
   Hier sei als Beispiel das Feld `ICA_CD` genannt.

Aufgrund der aufgeführten Unterscheidungsfälle ist es nun möglich, die Behandlung der einzelnen Merkmale bezüglich dieser Erkenntnisse auszurichten. Als wesentliches Kriterium kann die Entropiedifferenz aus legler und illegaler Entropie angesehen werden. Als Beispiel für den Einsatz der Entropiewerte sollen die Felder `MSG_TYP` und `TRN_TYP` dienen. Hier ist bei den illegalen Datensätzen eine relativ kleine Entropie festzustellen, jedoch bei den legalen eine höhere Entropie, die nahezu der totalen Entropie[14] entspricht. Es ist also daraus zu schließen, daß bei den Mißbrauchsdatensätzen in den entsprechenden Feldern *eher* spezifische Daten zu finden sind als bei den legalen Datensätzen. Diese spezifischen Daten gilt es dann bei der Generalisierung zu bewahren und nicht zu verallgemeinern.
Im vorgestellten Algorithmus wirkt sich die Entropiedifferenz jedoch speziell auf die Eigenschaft der Wildcards aus. Ist die Entropiedifferenz eines speziellen Feldes niedrig, das heißt unterschreitet sie eine Schwelle, so wird der Wildcard derart behandelt, daß nur *die* Werte von diesem Platzhalter abgedeckt werden, die tatsächlich in diesem Zusammenhang der Generalisierung vorkommen; der Wildcard deckt nur die Werte ab, die explizit zu der Verallgemeinerung an dieser Stelle beigetragen haben. Übersteigt die Entropiedifferenz die festgelegte Schwelle, so

[14] Entropie der Daten beider Klassen zusammen

wird der Wildcard allgemein behandelt und kann alle möglichen Werte annehmen, selbst die Werte, die von diesem Datum unter diesen Umständen gar nicht verwendet werden.

Ist die Konfidenz einer Regel nicht mehr ausreichend, so braucht diese Regel nicht weiter im Rahmen der Generalisierung beachtet werden, da sie die nötige Zuordnungszuverlässigkeit bezüglich des Mißbrauchs nicht besitzt. Auch für eine weitere Generalisierung ist diese Regel unbrauchbar, da nach Satz 1 gilt:

Satz 1 *Die Konfidenz c_{gen} einer generalisierten Regel erreicht höchstens die maximale Konfidenz c_{max} der Ursprungsregeln. Formal:*

$$c_{gen} = max(c_1, \ldots, c_n)$$

Beweis 1 *Die Konfidenz ist der Quotient aus Häufigkeit v des Regelrumpfes X in Verbindung mit dem Mißbrauch Y und der Gesamthäufigkeit w des Regelrumpfes X.*

$$c_i = \frac{X_i \cup Y_j}{X_i} = \frac{v_i}{w_i} = \frac{r_{ij}}{r_{i0} + r_{i1} + \cdots + r_{ip}}$$

O.B.d.A. gilt also:

$$c_1 = \frac{v_1}{w_1} \quad und \quad c_2 = \frac{v_2}{w_2} = \frac{\alpha v_1}{\beta w_1} \quad mit \quad \alpha, \beta > 0 \quad und \quad c_1 \geq c_2$$

Weiter gilt:

$$c_{gen} = \frac{v_1 + v_2}{w_1 + w_2}$$

Daraus folgt:

$$\alpha \leq \beta \Leftrightarrow 1 \geq \frac{\alpha}{\beta} \Leftrightarrow 1 \geq \frac{1+\alpha}{1+\beta} \Leftrightarrow c_1 \geq \frac{(1+\alpha)v_1}{(1+\beta)w_1} \Leftrightarrow c_1 \geq \frac{v_1 + v_2}{w_1 + w_2} = c_{gen}$$

Das heißt, die Konfidenz der generalisierten Regel c_{gen} ist immer kleiner als die größere Konfidenz der beteiligten Ursprungsregeln, in diesem Fall c_1. Da die Wahl der Regeln R_i mit Konfidenz c_i beliebig aber fest ist, kann diese Konfidenzhierarchie für beliebig (n) viele Ursprungsregeln gezeigt werden. Somit gilt die Behauptung. □

In Anlehnung an Satz 1 gilt weiter:

Satz 2 *Die Abdeckung s_{gen} einer generalisierten Regel ist mindestens so groß wie die Abdeckungen der Ursprungsregeln. Formal:*

$$s_{gen} \geq max(s_1, \ldots, s_n)$$

Beweis 2 *Die Abdeckung ist der Quotient aus Auftrittshäufigkeit der Mißbrauchsregel R_{ij}, also r_{ij} mit $j > 0$, und der Gesamtzahl aller Mißbrauchsregeln, der Menge $\mathcal{M}$.*

$$s_n = \frac{\|R_M\|}{\|\mathcal{M}\|} = \frac{v_n}{\|\mathcal{M}\|}$$

O.B.d.A. gilt also:

$$s_1 = \frac{v_1}{\|\mathcal{M}\|} \quad und \quad s_2 = \frac{v_2}{\|\mathcal{M}\|} = \frac{\alpha v_1}{\|\mathcal{M}\|} \quad mit \quad \alpha, v_n > 0 \quad und \quad s_1 \geq s_2$$

Weiter gilt:

$$s_{gen} = \frac{v_1 + v_2}{\|\mathcal{M}\|}$$

und weiter

$$s_{gen} \geq s_1 \quad und \quad s_{gen} \geq s_2$$

auch hier gilt, durch die beliebige Wahl der Abdeckungsergebnisse kann das Ergebnis auf alle Kombinationen aus Regeln erweitert werden, und gilt somit für die sämtliche Regeln $R_1, \ldots, R_n$ mit den Abdeckungswerten $s_1, \ldots, s_n$. □

Eine neu generalisierte Regel wird mit den vorhandenen, potentiellen Generalisierungsregeln verglichen, so daß keine doppelten Regeln und somit Redundanzen erzeugt werden.
Konnte im Zuge einer Zusammenführung zweier Regeln keine neue Regel verbucht werden, da die Konfidenz- und Abdeckungsschwellenwerte nicht überschritten werden konnten, so wird dieser ergebnislose Versuch in Zusammenhang mit den beteiligten Regeln verbucht, um den selben Vergleich später zwischen den Regeln auch im Zuge einer Baumsuche zu vermeiden. Diese Vergleichslisten werden gelöscht, sobald eine Regel als „gebraucht“ markiert, also zu einer Generalisierung benutzt wurde und damit aus der Liste der potentiell zur Generalisierung verwendeten Regeln entfernt wird. Damit können langwierige Regelvergleiche gespart und somit der Algorithmus beschleunigt werden.

Die Suche nach Alternativregeln

Bei der Generalisierung werden soweit wie möglich ähnliche Regeln zusammengefügt. Dabei kann es natürlich passieren, daß eine neue Regel auf einem bestimmten Abstraktionsniveau bei der Zusammenlegung aus den zwei Ursprungsregeln R^1 und R^2 die festgelegte Konfidenzhürde nicht erreicht. Ist nun aber die Abdeckung dieser neuen Regel ausreichend hoch – mit fortschreitendem Generalisieren nimmt die Abdeckung monoton zu (siehe Satz 2)–, so wird versucht, dennoch eine Regel zu konstruieren, die jedoch auf den weniger stark generalisierten

Vorgängerregeln von R^1 und R^2 basiert und in Folge dessen die Konfidenzschwelle erreicht wird.
Eine solche Regel kann durchaus existieren, da zu einem früheren Generalisierungszeitpunkt der festgelegte Regelabstand α eine Zusammenlegung nicht zugelassen hat. Erst durch fortschreitendes Verallgemeinern der einzelnen Regeln kann es später zu einer solchen Zusammenlegung der Regeln kommen. In Abbildung 3.6 ist ein typisches Beispiel für eine derartige Situation dargestellt.

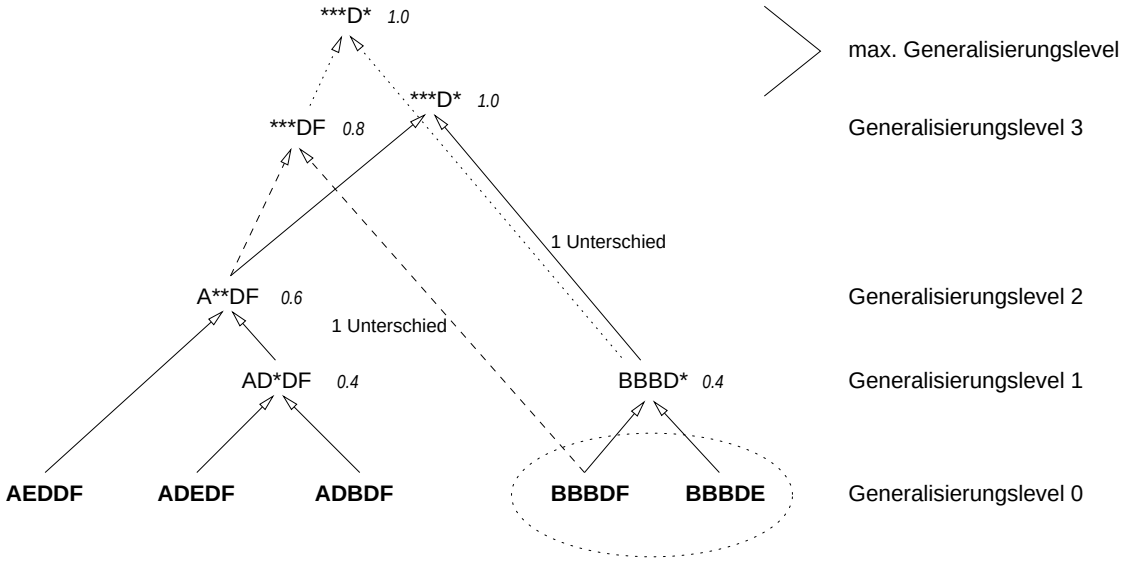


Abbildung 3.6: Generalisierungsgraph (Beispiel: Baumsuche)

Auf dem Generalisierungslevel 0, den Blättern des Generalisierungsbaumes, befinden sich die Orginalregeln, die noch keine Wildcards enthalten. Die kursiv gesetzten Zahlenwerte bezeichnen die jeweilige Abdeckung[15], die durch die entsprechende Regel gegeben ist. Im Falle der Regeln `A⋆⋆DF` und `BBBD⋆` kann es nun zu einem solchen Fall, der eine derartige Baumsuche notwendig macht, kommen. Je nach Konfidenzschwelle, kommt es zu einer Baumsuche, bei der die Unterbäume der aktuell betrachteten Regeln als Regelmenge für eine ausweichende Generalisierung hinzugezogen werden. Bei der Suche im rechten Teilbaum (gestrichelt umrandet) kann mit der Regel `BBBDF` eine neue Regel generalisiert werden, die eine höhere Abdeckung $(0,8)$ als die Ursprungsregeln $(0,6$ und $0,4)$ und außerdem eine ausreichende Konfidenz[16] besitzt.

Aufgrund der Voraussetzungen ist zu vermuten, daß bei erfolgreicher Suche in dem entsprechenden Unterbaum eine Regel gefunden wird, die auf jeden Fall eine

[15] Die Abdeckung in Bezug zu den Ursprungsregeln auf GL 0 in der Zeichnung

[16] Dies ist natürlich von der Anzahl der in diesem Zusammenhang betrachteten legalen Regeln und der Konfidenzschwelle abhängig.

größere oder gleiche Abdeckung und Konfidenz hat als die ursprünglichen Regeln, da in diesem Fall eine Regel eines ursprünglich, kleineren Generalisierungslevel rekrutiert wird und somit die Konfidenz angehoben werden kann. Diese Tatsache basiert auf der Eigenschaft, daß, wie in Satz 1 gezeigt, die zusammengesetzte, neue Regel höchstens die maximale Konfidenz der beiden Ursprungsregeln besitzen kann. Steigt also die Konfidenz einer Ursprungsregel, so ist damit die Möglichkeit gegeben, daß auch die Konfidenz der zusammengefügten Regel steigt. Desweiteren gilt, daß die Abdeckung mit der Vereinigung zweier Regeln aufgrund Satz 2 nur monoton zunehmen kann, so daß die Intention der Generalisierung trotzdem weiter verfolgt wird.

Bei einer derartigen Baumsuche werden ebenfalls wie bei dem eigentlichen Regelvergleich die zugrundeliegenden Parameter wie akuelle Distanzmaß und die Konfidenzschwelle berücksichtigt. Dazu muß jede potentielle neue Regel auf dieselbe Weise untersucht werden, wie die ursprünglichen Regeln im eigentlichen Generalisierungslauf (siehe Algorithmus C.1) auch.

3.10 Zusammenfassung und Rückblick

Das beschriebene und auch realisierte Assoziativspeichermodell entspricht dem Ansatz aus Abschnitt 3.4. Konträr zu einem gewöhnlichen Assoziativspeichermodell werden jedoch, um den Eingaberaum einzuschränken, die symbolischen Mißbrauchsdatentupel durch den Generalisierungsschritt verallgemeinert. Dieser reduzierte Eingaberaum wird, wie es die zweite Schicht vorsah, im gleichen Schritt entsprechend der Assoziationen mit Mißbrauch oder nicht Mißbrauch, statistisch auf Grundlage des Bayes-Kriteriums – der Konfidenz – ausgewertet. Regeln, die nicht mit ausreichender bedingter Wahrscheinlichkeit einem Mißbrauch zugeordnet werden können, brauchen nicht weiter betrachtet werden. Gleiches gilt für Regeln, die derart speziell sind, daß sie nicht mit einer ausreichenden relativen Häufigkeit – der Abdeckung – in den Mißbrauchsdaten vertreten sind. Dadurch verschiebt sich die Entscheidungsschwelle bezüglich der Ausgabe, da nur noch *die* Regeln behandelt werden, die eine ausreichende statistische Grundlage haben. In Anlehnung an Gleichung 3.24 gilt nun:

$$M(\mathbf{x}) = \begin{cases} -1 & \text{Mibrauch} \\ +1 & \text{kein Mißbrauch} \end{cases} \qquad \begin{array}{l} \sum_{k=1}^{p} M_k(\mathbf{x}) > 0 \\ sonst \end{array} \tag{3.26}$$

Damit erübrigt sich im eigentlichen Sinne eine derartige Schwellenfunktion, da nun einfach bei Regelübereinstimmung für Mißbrauch entschieden werden kann.

3.11 Implementierung

3.11.1 Voraussetzungen für Data-Mining-Verfahren

Allgemeine Leitlinien

Um die Daten mit Hilfe eines Data-Mining-Verfahrens bearbeiten zu können, sind bestimmte Voraussetzungen der Daten einzuhalten. So müssen verschiedene Anforderungen an die zugrundeliegenden Daten erfüllt sein, um das statistisch fundierte Arbeiten auf diesen Daten zu erlauben. Im Zweifelsfall müssen die Daten dahingehend bearbeitet und konvertiert werden. In [Kim97] sind die wesentlichen Punkte kurz zusammengefaßt. Da beim Data-Mining Übereinstimmungen zwischen den Daten gesucht werden, so daß allgemeine Regeln daraus abgeleitet werden können, muß von klassifizierten, also geklusterten oder symbolischen Daten, ausgegangen werden. Analoge Daten würden zu derart differenzierten Regeln führen, daß eine Datenanalyse aufgrund dieser Regeln nicht sinnvoll wäre. Ebenso verhält es sich bei Schlüsseln, die in fast allen Datenbanken vertreten sind, um Datensätze eindeutig zu markieren. Gerade diese Eindeutigkeit führt beim Data Mining zu Fehlverhalten, denn hier läßt sich *keine* generelle Regel finden!

`NULL`-Werte

Auch sogenannten `NULL`-Werten muß eine differenzierte Beachtung geschenkt werden. Hiebei handelt es sich um nicht gefüllte Datenfelder eines Datensatzes. Bei eigentlichen Data-Mining-Verfahren werden die kompletten Datensätze entweder unberücksichtigt gelassen oder mit dem wahrscheinlichsten der Werte oder einem Mittelwert gefüllt. Diese Felder beinhalten keine eindeutige Aussage über das entsprechende Datum.
Im vorliegenden Fall hingegen wird davon ausgegangen, daß gerade durch das Fehlen dieser Informationen eine Aussage möglich ist, da diese `NULL`-Werte in den vorliegenden Daten möglicherweise ebenfalls Rückschlüsse auf die Transaktion beziehungsweise deren Buchungshergang zulassen. Aus diesem Grund wurden sie regulär, wie auch die übrigen symbolischen Daten, abgearbeitet und einem numerischen Wert zugeordnet. In diesem Zusammenhang ist zum Beispiel denkbar, daß eine Transaktion an einem mißbräuchlich modifizierten Terminal getätigt wurde, von dem in Folge des Eingriffs nicht alle Daten komplett übermittelt werden – es kommt zu `NULL`-Werten in den Transaktionsdaten. Aber auch legale Transaktionen können `NULL`-Werte enthalten, wenn eine entsprechende Zusatzinformation nicht vorrätig oder überflüssig, beziehungsweise optional ist.

3.11.2 Vorbereitung zur Generalisierung

Um die meist alpha-numerisch kodierten Werte in ein einfaches Zahlenformat umzuwandeln, wurde jedem neuen Wert eine individuelle Zahlenkodierung zuge-

ordnet, so daß eine eineindeutige Zuweisung erfolgen kann. Die so entstandenen Übersetzungstabellen sind ebenfalls in der zentralen Datenbank unter dem Namen der jeweiligen Spaltennummer und -bezeichung abgelegt. Damit ist jederzeit auch eine Rückumwandlung in die ursprünglichen, alpha-numerischen, symbolischen Werte möglich.

Für den eigentlichen Generalisierungslauf wurde zunächst nur ein Teil der legalen Transaktionsdatensätze in das numerische Format übersetzt. Es wurden zunächst 30000 Datensätze umgewandelt. Dabei wurden durch eine zufällig bestimmte Schrittweite im Intervall I mit $I = \{1, \ldots, 18\}$[17] die einzelnen legalen Datensätze aus der Datenbank extrahiert und konvertiert. Auf diese Weise ist sichergestellt, daß ein möglichst allgemeiner Querschnitt der legalen Daten in der Teilauswahl enthalten ist. Von den 5850 als illegal ausgezeichneten Datensätzen wurden alle kodiert und zur Generalisierung der Regeln verwendet. Die zur Generalisierung verwendeten Konfidenz- beziehungsweise Abdeckungswerte werden auf Basis dieser Regeln in den jeweiligen Regelteilmengen bestimmt. Um im Falle der Konfidenz den Teil der legalen Daten auf ein reales Verhältnismaß hochzurechnen, wird die bestimmte Anzahl mit dem Hochrechnungsfaktor h (siehe Gleichung 3.25 auf Seite 41) multipliziert.

Desweiteren wurden die unter anderem für die Generalisierung verwendeten Entropiewerte auf allen Daten bestimmt. Dazu wurde für jeden Wert aus den Übersetzungstabellen die Auftrittswahrscheinlichkeit berechnet, und entsprechend Gleichung 3.6 auf Seite 22 zu den entsprechenden Entropiewerten der einzelnen Datentypen verrechnet. Als Beispiel diene hier der Kartentyp `CRD_TYP`. Dieses Feld enthält in den vorliegenden Daten die in Tabelle 3.1 aufgeführten Auftrittshäufigkeiten sowie -wahrscheinlichkeiten.

Bezeichnung	Anzahl			Wahrschein-
	legal	illegal	gesamt	lichkeit (%)
`NULL`	155	-	155	0,028
EM	542703	5850	548553	99,972

Tabelle 3.1: Auftrittshäufigkeit und Wahrscheinlichkeiten der `CRD_TYP` Werte

In Tabelle 3.1 ist die Gesamtwahrscheinlichkeit, die zur Berechnung der Gesamtentropie herangezogen wird, aufgeführt; die einzelnen klassenspezifischen Wahrscheinlichkeiten berechnen sich entsprechend der Mächtigkeiten der einzelnen Datenklassen.

[17] Damit kann maximal der gesamte Datenbereich von 542858 Datensätzen regelmäßig durchschritten werden.

3.11.3 Umsetzung und Ausführung

Der eigentliche Algorithmus zur Generalisierung, wie er auch abstrahiert in Anhang C in Form von Pseudokode aufgeführt ist, wurde aus Effizienzgründen in C++ implementiert. In diesem Fall wird Maschinencode ausgeführt. Eine Implementierung in JAVA mußte ausgeschlossen werden, da im Gegensatz zu C++ JAVA eine interpretierte Sprache ist, die von einer so genannten „JAVA VIRTUAL MACHINE“ ausgewertet wird und bei weitem nicht die Arbeitsgeschwindigkeiten von Maschinencode erreicht. Dies macht sich besonders bei den vielen Vergleichsdurchläufen bemerkbar. Es ist von einer um das 20-fache verminderten Arbeitsgeschwindigkeit im Vergleich mit Maschinenkode die Rede [Gem99].

Laufzeitabschätzung

Für die Laufzeitabschätzung wird im wesentlichen die Anzahl der Vergleiche in Abhängigkeit der zu generalisierenden Mißbrauchsregeln n betrachtet. Da alle Regeln der Reihe nach mit den entsprechend nachfolgenden verglichen werden, so daß kein Vergleich doppelt ausgeführt wird, ergibt sich für die Anzahl an Vergleichen:

$$\#\ \text{Vergleiche} = \sum_i i = \frac{n \cdot (n-1)}{2} \tag{3.27}$$

Dies entspricht einer quadratischen Laufzeit, oder in *O-Notation*, einer Laufzeit von $O(n^2)$. Jedoch dürfen ab einem bestimmten Zeitpunkt die Baum-Suchläufe nicht außer acht gelassen werden. Die Anzahl der Vergleiche, die hier getätigt werden müßsen, hängt von der Anzahl der inneren Knoten in dem entsprechenden Unterbaum ab. Abhängig von der Anzahl der Blätter b des Unterbaumes kann so die Anzahl an inneren Knoten abgeschätzt werden [OW90]. Es gilt maximal für die Anzahl an Vergleichen bei der doppelten Baumsuche:

$$\#\ \text{Vergleiche} = 2 \cdot \big((2 \cdot b - 1) - 1\big) \tag{3.28}$$

Die Wurzel darf als innerer „Vergleichsknoten“ nicht mit gezählt werden. Es ist weiterhin zu berücksichtigen, daß jeweils beide Unterbäume der zu vergleichenden Regeln durchsucht werden müssen, und somit der Faktor 2 zusätzlich wie in Gleichung 3.28 mit einkalkuliert werden muß. Im schlimmsten Fall kann die Anzahl der Blätter eines Unterbaumes $n-1$ Blätter betragen.
Damit kann also generell eine Laufzeit von $O(n^3)$ abgeschätzt werden, geht man davon aus, daß im „worst case“ bei einem Gros der Regelvergleiche sich eine Baumsuche anschließt. Dies gilt gerade in Anbetracht der Tatsache, daß mit abnehmender Regelanzahl die Konfidenz sinkt und umgekehrt dazu die Abdeckung der einzelnen verallgemeinerten Regeln steigt, und damit die Baumsuchen immer öfter ausgeführt werden. Zudem wachsen die Regelbäume mit fortgesetzter

Generalisierung und damit nimmt die Anzahl der Blätter beziehungsweise der inneren Knoten zu. Es kann also bezüglich der Laufzeit gesagt werden, daß diese im Verlauf der Generalisierung von quadratischer in kubische Laufzeit übergeht. Diese Laufzeitverschlechterung muß als Preis für die genauere, optimierte Regelgeneralisierung in Kauf genommen werden.

3.11.4 Programmbeschreibung

Die Implementierung der Generalisierung unterteilt sich in nur wenige Klassen und Programmabschnitte. So wurde der eigentliche Generalisierungsprozeß in der Datei `genRules` beziehungsweise `jgenRules` für die Anbindung an ein JAVA -Interface implementiert. Hier finden auch die Baumsuchläufe statt. Für eine Regelgeneralisierung können die Benutzerparameter aus Tabelle 3.2 vorgegeben werden:

Parameter	Beschreibung
Datenbasis	Dateinamen der kodierten, legalen und illegalen, symbolischen Transaktionsdaten
Mindestkonfidenz	Minimale Konfidenzschranke; liegt die Konfidenz einer Regel unterhalb dieses Wertes, wird diese Regel verworfen und nicht mit Mißbrauch assoziiert.
Mindestabdeckung	Minimale Abdeckungsschranke; kann diese Mindestabdeckung von einer generalisierten Regel nicht erreicht werden, wird sie nicht als solche in die Liste der potentiell zu generalisierenden Regeln aufgenommen, sondern verworfen.
Mindestsupport	Diese Schranke spielt nur eine untergeordnete Rolle und wird für die aktuellen Betrachtungen nicht verwendet.
Generalisierungsschritte	Anzahl der Generalisierungsschritte, die maximal durchgeführt werden sollen.
Entropieschwelle	Dieser Wert bestimmt die Einteilung der Wildcards. Liegt eine Entropiedifferenz (siehe auch Abschnitt 3.2) vor, die kleiner ist als der gewählte Wert, so werden von dem Wildcard nur die bei der Zusammenführung der Regel beteiligten Werte abgedeckt, ansonsten alle.
Samplegrößen	Diese bestimmen die Größen der zur Generalisierung verwendeten, numerisch kodierten Teilmengen der vorhandenen Datensätze.

Tabelle 3.2: Generalisierungsparameter

Die Parameter im unteren Teil der Tabelle sind „hart kodiert“, also in das Pro-

gramm hineinprogrammiert, da es sich um feste, vorgegebene Werte handelt, die nur im Bedarfsfall geändert werden sollen.
Desweiteren wurden folgende Klassen implementiert:

`Rule_C:` Diese Klasse enthält alle regelrelevanten Methoden und speichert die Werte der entsprechenden Regeln sowie die zugehörigen Statistikwerte wie die absoluten Auftrittshäufigkeiten, die Konfidenz und die Abdeckung. Zu den implementierten Methoden zählen verschiedene Konstruktoren, als auch Zugriffsmethoden auf die Regel selbst und andere regelspezifische Eigenschaften. Außerdem wird hier in der Methode `mergeRules` die Zusammenfügung zweier Regeln durchgeführt sowie der Abstand zwischen zwei Regeln mit `ruleDiff_W` beziehungsweise `ruleDiff` bestimmt.

`dmRules:` In dieser Klasse sind alle Methoden zur Bestimmung der statistischen Regeleigenschaften implementiert. Hier wird die Menge der Vergleichsregeln verwaltet, zum Beispiel mit `fillList` eingelesen. Außerdem werden hier die Entropiewerte in `sortEntropien` in Beziehung zu der Vergleichsreihenfolge der einzelnen Daten gesetzt.

`Liste:` Dies ist eine dynamische Liste für die Verwendung der Regeln mit allen zugehörigen Funktionen wie `insert`, `insertSorted` und `getListLength`. Außerdem wurden Ausgabefunktionen wie etwa `printList` oder Zugriffsroutinen implementiert wie zum Beispiel `getNextItem`.

`Reader_class:` Diese Klasse ist zuständig für den Datenaustausch zwischen dem Programm und den in den Dateien abgespeicherten Regeln. Hier werden die gespeicherten Regeln formatiert eingelesen. Diese Methode wird unter anderem auch dazu verwendet, um Zusatzinformationen wie zum Beispiel den Konfidenzwert `getAktKonf` für die spätere graphische Ausgabe aus den Dateien mit den generalisierten Regeln auszulesen.

3.12 Ergebnisse

In diesem Abschnitt sollen die Ergebnisse der Generalisierung der Mißbrauchsdatensätze vorgestellt werden. Es soll versucht werden, die Ergebnisse aus verschiedenen Blickwinkeln zu untersuchen und Aussagen über die gefundenen Regeln zu geben. In diesem Zusammenhang werden die gebildeten Regelsätze charakterisiert und auf ihre Brauchbarkeit zur Einschränkung des Mißbrauchsraumes hin untersucht. Der Abschnitt unterteilt sich in mehrere Sektionen; zunächst werden die verwendeten Parameter, die zur Generalisierung verwendet wurden vorgestellt (Abschnitt 3.12.1), weiter werden auszugsweise die gebildeten Regeln vorgestellt (Abschnitt 3.12.2), und die Eigenschaften aller produzierten Regeln mit Hinblick unter anderem auf die Data-Mining-Werte wie Gesamtkonfidenz oder -abdeckung sowie auf den Umfang der gebildeten Regelmenge (Abschnitt 3.12.3) ausgewertet.

3.12.1 Generalisierungsparameter

Zur Generalisierung wurden 5850 Mißbrauchsdaten in das numerische Format umgewandelt und verwendet. Von den 542858 legalen Datensätzen wurden hingegen zunächst nur 30000 Datensätze konvertiert und für den Generalisierungsschritt benutzt. Es stellte sich jedoch in einem anschließenden Vergleich der auf dem Sample erzielten Ergebnisse mit den auf den gesamten vorliegenden Daten erreichten heraus, daß eine solche Teilmenge scheinbar nicht ausreichend repräsentativ die gesamte Spannbreite der legalen Datensätze abdecken kann. Aus diesem Grunde wurde ein weiteres Sample im Umfang von 30000 Datensätzen zur Generalisierung herangezogen. Dabei wurde erneut versucht, möglichst den gesamten Raum der legalen Datensätze stichprobenhaft als Grundlage dienen zu lassen. Durch eine veränderte Schrittweite konnten so andere Daten aus der Menge der gesamten legalen Daten als bei der ersten Samplegenerierung ausgewählt werden. Die generalisierten Regeln auf Basis dieser legalen Regeln erreichten wiederum nur im Vergleich zur Gesamtdatenmenge stark abweichende Statistikwerte wie Konfidenz oder Abdeckung. Auch der Vergleich der Teilmengen untereinander bestätigte diese Erkenntnis. Aus diesem Grund wurden die Regelmengen zusammengelegt zu einer Teilmenge von 60000 Regeln. Mit dieser Menge konnte letztendlich eine akzeptable Übereinstimmung mit der Gesamtdatenmenge bezüglich der bestimmten Konfidenz- und Abdeckungswerte erzielt werden, so daß die Generalisierung sinnvoll eingesetzt werden konnte.

Laut Aussage des Kreditinstituts ist eine Vertrauensgrenze bezüglich der Mißbrauchszuweisung für das Auftreten des Regelrumpfes der vorliegenden Regel in Zusammenhang mit einem Mißbrauch von 10% vertretbar. Ein direktes Einschreiten kann jedoch von weiteren Faktoren wie zum Beispiel der Transaktionshistorie abhängen. Umgangssprachlich bedeutet das, daß bei einer Regel, die in 10 von 100 Fällen als Mißbrauch auftritt, die Aufmerksamkeit gesteigert und mit Hilfe

weiterer Analyseschritte die Beurteilung überprüft und präziser gestaltet wird. Aus diesem Grunde wurde eine Konfidenzschwelle von 20% gewählt, um im Anbetracht der verwendeten Teilmenge und dem nur ca. hundertsten Bruchteil des als real angenommenen Datenumfangs auf Seite der legalen Daten einen Puffer bei der Bestimmung der relevanten Data-Mining-Werte vorzusehen.

Der Abdeckungswert wurde sehr niedrig angesetzt, um auch spezialisiertere Mißbrauchsregeln aus der Menge herauszufiltern. Die Mindestabdeckung wurde auf einen Wert von 0,2% gesetzt. Das heißt, daß eine Regel in die potentiellen, verallgemeinerten Mißbrauchsregeln aufgenommen wird, wenn sie mindestens 12 mal[18] in der gesamten Menge der Mißbrauchsdaten vorkommt.

Der Support einer Regel spielt in etwa die gleiche Rolle wie die Abdeckung, ist jedoch nicht auf die Mißbrauchsdaten beschränkt und hat somit in dem Zusammenhang mit der Verallgemeinerung der Mißbrauchsregeln keine Bedeutung. Aus diesem Grund wurde dieser Wert auf 0 gesetzt und nicht weiter beachtet.

Zuletzt sei noch die Entropiegrenze erwähnt. Dieser Wert entscheidet über die Behandlung der Wildcards, den Platzhaltersymbolen, bei der Generalisierung. Liegt der entsprechende Entropiewert des aktuellen Datentypus unter dieser Entropieschwelle, so werden im Zuge der Generalisierung von den Platzhaltern *nur die* Werte abgedeckt, die zu dem Wildcard geführt haben. Kann diese Entropieschwelle nicht unterschritten werden, so wird der Wildcard als universeller Platzhalter angesehen und deckt in Folge dessen sämtliche vorkommenden Werte ab. Leider ist diese differenzierte Behandlung nur im Rahmen der Generalisierung an sich möglich, ohne den Umfang an Verwaltungsaufwand zu sprengen und die spezifischen Werte, die in Verbindung mit den jeweiligen Wildcards gebraucht werden, für spätere Verwendungen zu speichern.

3.12.2 Generalisierungsergebnisse

Die Generalisierung der Mißbrauchsdatensätze benötigte auf einem PENTIUM II - 233 Mhz (128 Megabyte Hauptspeicher) je nach verwendeten Parametern für 700 Iterationen ca. 12 Stunden. Für diese Laufzeit sorgten vor allem die Baumsuchen im fortgeschrittenen Stadium der Generalisierung.

Mit den in Abschnitt 3.12.1 vorgestellten Parametern, zusammengefaßt in Tabelle 3.3, konnten Regeln bis zu einem Generalisierungslevel (GL) von 17 erzeugt werden. Das heißt, die allgemeinsten Regeln besitzten 17 Wildcards. Diese Regeln des GL 17 sind in Tabelle 3.4 aufgeführt.

[18] 0,2% von 5850 = 11,7

Parameter	Wert
Eingabedaten	60000 legale, symbolische Daten inkl. Kontonummer 5850 illegale, symbolische Daten inkl. Kontonummer
Mindestkonfidenz	0,2 = 20%
Mindestabdeckung	0,002 = 2%
Mindestsupport	0
Entropieschwelle	1,0
Regellänge	27
Generalisierungsschritte	700

Tabelle 3.3: Generalisierungsparameter

Regel	ACCT_NBR	TRN_TYP	CURR_CD	POS_ENT_CD	FAL_SCOR	CRD_TYP	ICA_CD	AID_CD	SIC_CD	ACT_CD	MSG_TYP	MER_ID	MER_CNTY_CD	CTY_1
1	⋆	EA	840	⋆	⋆	EM	2264	8402646	⋆	⋆	1100	null	000	⋆
2	⋆	EA	840	⋆	995	EM	⋆	⋆	⋆	000	1100	⋆	000	⋆

Regel	POST_CD_1	CNTY_CD_1	CR_LMT	ACTV_IND	ACCT_STAT	CTY_2	POST_CD_2	ADDR_STAT	EMIT_NBR	INST_NBR	ISS_REAS	GEN_CD	CARD_TYP
1	⋆	⋆	⋆	I	⋆	⋆	⋆	null	⋆	⋆	⋆	⋆	⋆
2	⋆	⋆	⋆	I	F8	⋆	⋆	null	⋆	⋆	⋆	⋆	⋆

Tabelle 3.4: Regeln aus Generalisierungslevel 17

Alle übrigen, generalisierten Regeln sind zusammengefaßt in den einzelnen Generalisierungsleveln im HTML-Format auf der beiliegenden CD-Rom aufgeführt. Die zugehörigen Konfidenz-, Abdeckungs- und Supportwerte auf Basis des verwendeten, legalen Datensample sind ebenfalls mit den Regeln verzeichnet.

3.12.3 Regelauswertung

Betrachtet man die generalisierten Regeln als reduzierte Mißbrauchsregelmenge, so lassen sich interessante Informationen und Charakterisierungen der Menge herleiten. Anhand der graphischen Darstellung können verschiedene Eigenschaften der Regelmenge veranschaulicht werden.

In Abbildung 3.7 sind verschiedene Abdeckungswerte abgetragen. Aus der Graphik ist ersichtlich, daß inklusive der Regeln aus dem GL 5 eine Gesamtab-

deckung, die Abdeckung aller Regeln bis einschließlich diesem GL vom maximalen GL 17 ab gesehen, von über 85% erreicht wird. Die Anzahl der für diese Abdeckung erforderlichen Regeln beträgt 633 Regeln, also gut ein zehntel der gesamten Mißbrauchsdatensätze. Fügt man Regeln kleinerer GL an, so kann kein wesentlich größerer Zugewinn bezüglich der Abdeckung verzeichnet werden, die Steigung der Kurve nimmt im letzten Viertel der GL (GL 4 - 0) ab.

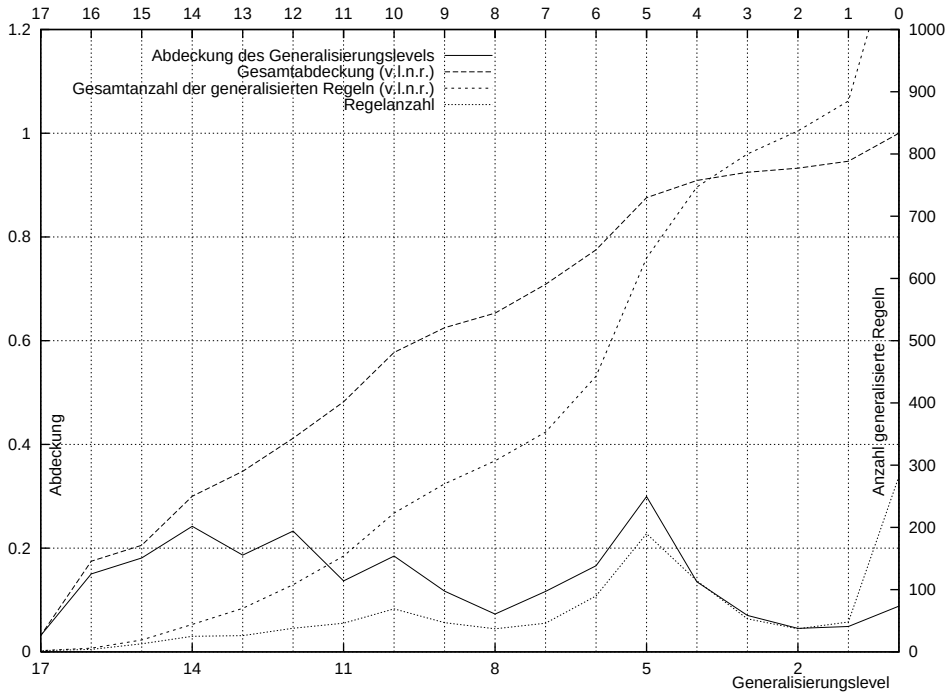


Abbildung 3.7: Abdeckungswerte der generalisierten Regeln

Auch ist in Abbildung 3.7 ersichtlich, daß die Abdeckung durch Hinzunahme weiterer Regeln nur monoton zunehmen kann. Durch die Gesamtanzahl *aller* generalisierter Regeln in Verbindung mit den noch nicht generalisierten Regeln wird, wie zu Beginn, eine Abdeckung aller Mißbrauchsregeln von 100% erzielt. Weiter läßt sich anhand der Graphik beobachten, daß mit abnehmendem Generalisierungslevel das Verhältnis aus Levelabdeckung und Regelanzahl abnimmt. Das bedeutet, je weniger die einzelnen Regeln der verschiedenen GL verallgemeinert sind, je weniger Transaktionsdatensätze werden im Verhältnis zur Regelanzahl pro GL abgedeckt.
In Abbildung 3.8 sind die Regeleigenschaften aus dem Blickwinkel der Konfidenz abgetragen. Kontrovers zu der festgelegten Mindestkonfidenz bei der Generalisierung verläuft die Konfidenz, die im Mittel für alle Regeln pro Level berechnet wird, leicht unterhalb dieser Schwelle. Die Ursache ist bei den Wildcards zu suchen, deren eventuelle Spezifität durch das Speichern der Regeln für die Auswertung bezüglich der Graphiken verloren gegangen ist. Wie in 3.12.1 schon erläutert, würde eine Verwaltung der Wildcardwerte über verschiedene Arbeitsschritte nicht

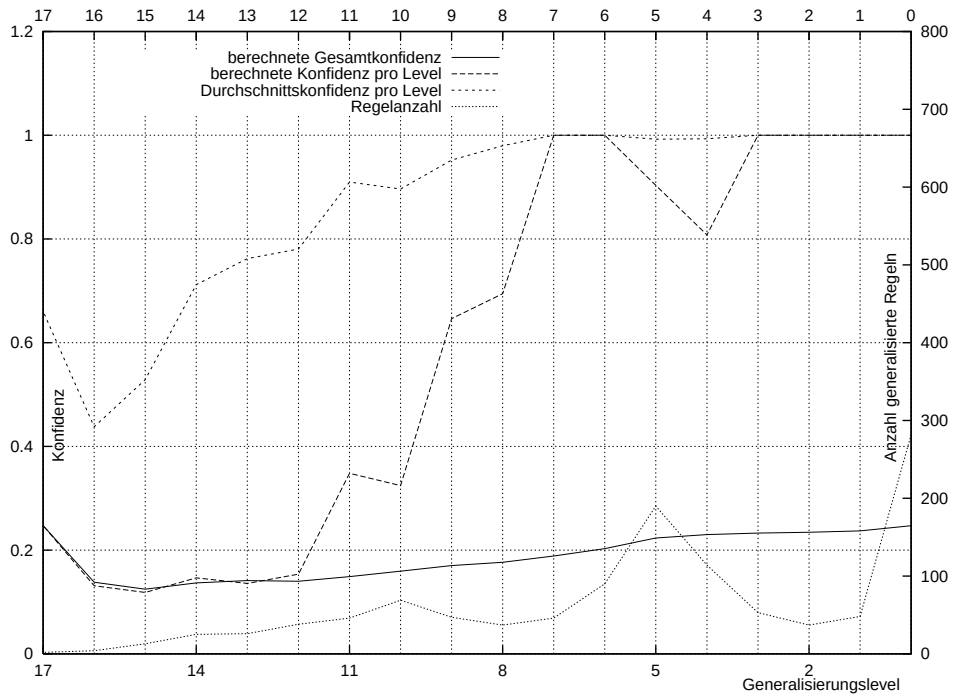


Abbildung 3.8: Konfidenzwerte der generalisierten Regeln

in Relation zum Aufwand stehen. Mit den Regeln des GL 11 übertrifft jedoch die Konfidenz die Mindstkonfidenzschwelle wieder. Die Durchschnittskonfidenz zeigt im Unterschied zur berechneten Konfidenz das arithmetische Mittel der bei der Generalisierung zu den Regeln bestimmten Konfidenzwerte auf den zur Generalisierung benutzten Teilmengen pro GL. Diese liegen deutlich über der festgelegten Mindestkonfidenzschwelle.
Mit dem GL 11 ist ein rapider Anstieg der berechneten Durchschnittskonfidenz zu verzeichnen. Die Konfidenz erreicht mit dem GL 7 zum erstenmal, vom GL 17 ausgehend die 100%-Marke. Das bedeutet, daß die Regeln dieser GL derart speziell die Mißbrauchsregeln vertreten, daß es zu keiner Übereinstimmung bezüglich der legalen Daten kommt.

Betrachtet man jedoch die Ergebnisse bezüglich der einzelnen Regeln auf allen vorliegenden Datensätzen, so ist festzustellen, daß die Konfidenzwerte im Vergleich zu den auf der Teilauswahl bestimmten variieren. Tabelle 3.5 macht dies anhand der Konfidenzwerte der Regeln aus GL 16 deutlich. In diesem Zusammenhang ist es denkbar, eine Selektion *der* Regeln durchzuführen, die unter Verwendung der allgemeinen Wildcards eine ausreichende Konfidenz auf den vorliegenden Daten erreichen. Zu der Elimination der Regeln mit unzureichender Konfidenz auf den vorliegenden Daten kommen desweiteren Regeln, die von stärker verallgemeinerten Regeln beinhaltet oder ebenfalls abgedeckt werden, ohne daß sie als Basis für diese fungiert hätten. In letztem Fall wäre nämlich schon bei der

Regel	legal	illegal	Konfidenz	
			alt	neu
1⋆	500	690	20,947	11,352%
2⋆	47	78	21,849	13,345%
3	64	267	32,364	27,910%
4	0	42	100,000	100,000%

Tabelle 3.5: Generalisierungslevel 16: Regeln wurden auf 60000 legale Testdatensätze, 5850 illegale Testdatensätze generiert. Anzahl bezüglich allen vorliegenden Daten bestimmt. Konfidenz mit einem Hochrechnungsfaktor von 10,776 berechnet.

Generalisierung eine solche Regel von der allgemeineren ersetzt und außer in Folge einer Baumsuche nicht mehr aktiv an der eigentlichen Generalisierung beteiligt worden. In Tabelle 3.5 sind zum Beispiel die Regeln mit einer nicht ausreichenden Konfidenz mit einem Stern (⋆) gekennzeichnet. Durch die Entfernung dieser gekennzeichneten Regeln nimmt die Gesamtkonfidenz zu, da zum einen durch weniger Regeln weniger legale Transaktionen falsch zugewiesen werden und zum anderen verliert die Gesamtabdeckung, da weniger illegale Transaktionen abgedeckt werden.
Die Gesamtkonfidenz entspricht erneut der Konfidenz, die auf Basis aller Regeln vom maximalen GL bis einschließlich dem aktuellen GL erreicht wird. Diese übertrifft nur langsam den Wert der Mindestkonfidenz. Hierbei spielen zwei Faktoren eine Rolle. Zunächst muß die Anzahl an richtig abgedeckten Mißbrauchsregeln zunehmen, außerdem darf gleichzeitig die Anzahl der durch die Mißbrauchsregeln abgedeckten legalen Datensätze, also der Fehlalarme, nicht zu stark weiter ansteigen. Weiter heißt das, daß sich der Nenner aus Gleichung 3.17 auf Seite 31 nur langsam an den Zähler anpaßt. Erst mit dem GL 6 wird die gewählte Konfidenzschwelle überschritten.
An diesem Beispiel tritt zu Tage, daß es sinnvoll ist, einen Mindestkonfidenzpuffer bei der Generalisierung mit einzuplanen, so daß Reserven in Form einer ausreichenden Konfidenz auch nach der Generalisierung auf Basis der unspezialisierten Wildcards gewährleistet sind.
Weiter läßt sich in Bezug zu Abbildung 3.8 sagen, daß die Regeln im GL 17 trotz des hohen Generalisierungsniveaus eine hohe Verläßlichkeit auf den Teilmengen, die zur Generalisierung verwendet wurden, erreichen.

Einen weiteren Einblick in die Beziehung aus Konfidenz und Abdeckung soll Abbildung 3.9 geben. Hier sind die Gesamtkonfidenz und -abdeckung gegeneinander in Bezug zur Anzahl der generalisierten Regeln aufgetragen. Auch hier wird deutlich, daß erst mit dem GL 6 die Gesamtkonfidenz den Wert der Mindestkonfidenz übersteigt. Außerdem wird erneut ersichtlich, daß mit den Regeln des GL 5 eine Abdeckung von über 85% erreicht werden kann, und das mit einer

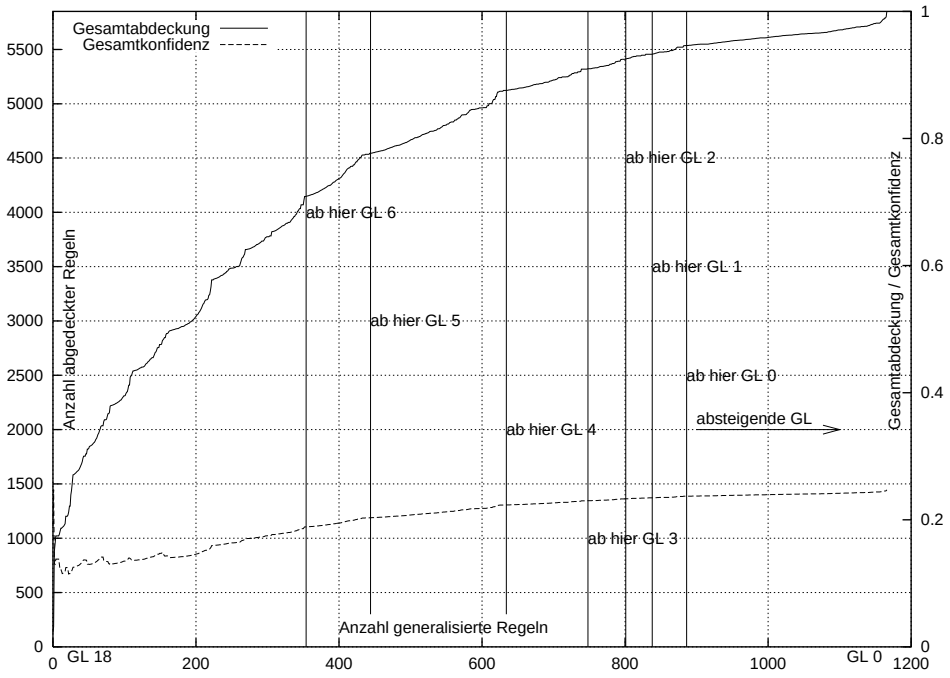


Abbildung 3.9: Gesamtkonfidenz und -abdeckungswerte auf 60.000 legalen und 5850 illegalen Datensätzen

Regelanzahl von 600 generalisierten Regeln. Für diese 10,26% Mißbrauchsregeln der eigentlich vorliegenden Mißbrauchsdatensätze kann also geschlossen werden, daß sie mit ausreichender Konfidenz – Verläßlichkeit – von über 20% mehr als 4000 Mißbrauchsfälle zuverlässig abdecken.

3.12.4 Regelauswertung auf allen Daten

Im Vergleich zu den bisherigen Auswertungen soll nun die generalisierte Regelmenge auf alle vorliegenden Daten angewandt werden. Dies ermöglicht eine genauere Analyse der verallgemeinerten Regeln und läßt eine realitätsnahe Interpretation zu.
Die Regeln werden entsprechend ihrer Bedeutung in das ursprüngliche Werteformat zurückkonvertiert und anschließend mit sämtlichen Transaktionsdatensätzen verglichen. Dabei spielt es eine Rolle, wie groß der Regelumfang gewählt wurde. Entsprechend den Erkenntnissen aus Abschnitt 3.12.3 und gemäß Abbildung 3.9 wurde zunächst versucht, mit den Regeln des GL 17 bis einschließlich GL 4, das sind insgesamt 747 Regeln, auch auf den vorliegenden Daten eine entsprechend hohe Abdeckung von über 80% zu erzielen.
Es wird nun jeder Datensatz mit sämtlichen ausgewählten Regeln verglichen. Kommt es zu *mindestens einer* Übereinstimmung zwischen dem aktuellen Datensatz und einer generalisierten Regel, so wird der entsprechende Datensatz mit

Mißbrauch assoziiert und als potentieller Mißbrauch angesehen. Auf diese Weise kann eine Statistik auf der Menge aller Daten ausgeführt werden. Die Ergebnisse lassen sich in Tabelle 3.6 ablesen.

Diagnose	legal	illegal	gesamt	Konfidenz
richtig	99,73% (541389)	90,91% (5318)	**99,64%** (546707)	25,146%
falsch	0,27% (1469)	9,09% (532)	0,36% (2001)	
gesamt	100% (542858)	100% (5850)	100% (548708)	

Tabelle 3.6: Abdeckungsergebnisse der 747 Regeln GL 17 - GL 4 auf sämtlichen vorliegenden Daten

Zunächst bestätigen sich die Ergebnisse in ihrer Gesamtheit. Es kann auch auf den vollständigen Daten – im Umfang entsprechend der letzten Zeile in Tabelle 3.6 – eine Konfidenz von 25,15% erreicht werden, außerdem übersteigt die Abdeckung auch hier die 85% Marke mit sogar 90.91%.

Schränkt man nun die Regelmenge der GL 17 bis GL 4, wie in Abschnitt 3.12.3 beschrieben, ein, so erhält man nur noch 510 generalisierte Regeln mit ausreichender Konfidenz auf den Generalisierungssamples. Die Abdeckungswerte bezüglich der gesamten vorliegenden Daten durch diese reduzierte Regelmenge sind in Tabelle 3.7 zusammengefaßt.

Diagnose	legal	illegal	gesamt	Konfidenz
richtig	99.97% (542709)	83.08% (4860)	**99,79%** (547569)	75,167%
falsch	0.03% (149)	16.92% (990)	0,21% (1139)	
gesamt	100% (542858)	100% (5850)	100% (548708)	

Tabelle 3.7: Abdeckungsergebnisse der 510 selektierten Regeln von GL 17 - GL 4

Zu Demonstrationszwecken soll an dieser Stelle die Abdeckung gemäß Abbildung 3.9 dieser eingeschränkten Mißbrauchsregelmenge abgetragen werden. Das Ergebnis ist in Abbildung 3.10 abgebildet. Es ist zu erkennen, daß die Gesamtabdeckung aller beteiligter Regeln auf den Mißbrauchsdaten abnimmt von 90,91% auf einen Wert von 83,08%.
Die Konfidenz kann bezüglich der Einschränkung der Mißbrauchsregeln nur zunehmen, da mit dem Entfernen einiger Regeln zwar auf der einen Seite weniger Mißbräuche aufgedeckt aber zum anderen auch weniger legale Datensätze fälschlicherweise abgedeckt werden. Da von diesem Faktum hauptsächlich die Regeln betroffen sind, die *keine ausreichende* Konfidenz auf den Gesamtdaten erlangen konnten, also vermehrt legale Daten statt illegale Daten abdeckten, steigt die Konfidenz um nahezu das dreifache auf 75,74% an.

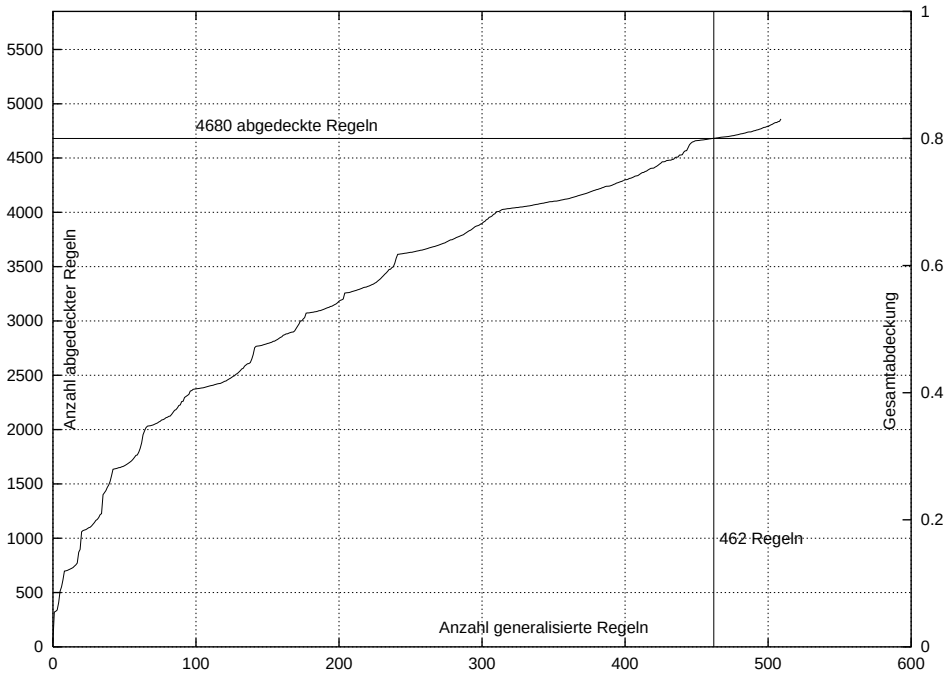


Abbildung 3.10: Gesamtabdeckung der bereinigten Regelmenge (510 Regeln) auf 60000 legalen und 5850 illegalen Datensätzen

Abschließend kann also zusammenfassend gesagt werden, daß mit 8,718% aller vorliegenden Mißbrauchstransaktionen eine Zuverlässigkeit bei der Mißbrauchszuordnung von bis zu 75,17% mit einer Abdeckung von 83,08% allein auf Basis der symbolischen Daten erreicht werden kann. Das bedeutet in Bezug auf die Mißbrauchsprävention, daß mit einer Menge von 510 Regeln 83% der bekannten Mißbräuche zugeordnet werden können. Es handelt sich also bei 8 allein auf diese Art und Weise deklarierten Mißbräuchen von 10 Fällen um tatsächliche Mißbräuche. Durch die Einschränkung der Regeln auf mehr als das 10-fache kann die Klassifizierung der beim Kreditinstitut eintreffenden Transaktionen auf Basis der symbolischen Daten um ein Wesentliches beschleunigt werden.

Kapitel 4

Auswertung analoger Daten mit Hilfe von radialen Basisfunktionsnetzen

Wie die Ergebnisse der Generalisierung zeigen, erreichen die Regeln bei hoher Abdeckung oft nur geringe Konfidenzwerte. Um die Konfidenz – die Zuordnungssicherheit – zu erhöhen, muß zusätzliche Information aus den Daten gewonnen werden. Eine ausschließliche Behandlung der symbolischen Daten würde zuviel Potential der kompletten Datensätze verschenken, auch wenn es sich dabei um das Gros der Daten handelt. Es ist also notwendig, möglichst sicher und effizient so viel Information wie möglich zusätzlich aus den analogen Daten zu beziehen. Dabei kommt es erneut darauf an, diese analogen Daten in kurzer Zeit den entsprechenden Mißbrauchskategorien fehlertolerant und doch mit ausreichender Genauigkeit, zuzuordnen.

In diesem Kapitel soll nun eine Methode vorgestellt werden, die es mit Hilfe neuronaler Netze ermöglicht, die Transaktionen bezüglich der Analogwerte einzustufen und zu klassifizieren. Dabei sollen zunächst die Anforderungen an ein solches System beschrieben werden (Abschnitt 4.1.1) und darauf basierend die Entscheidung für die Klassifizierungsmethode getroffen werden (Abschnitt 4.1.2). In Abschnitt 4.2 werden dann die Grundlagen und die Theorie der verwendeten Netzarchitektur vorgestellt. Anschließend folgt eine Beschreibung des implementierten Netzwerks (Abschnitt 4.3) und der verwendeten Algorithmen (4.4). Abschließend sollen die ermittelten Ergebnisse vorgestellt und diskutiert werden (Abschnitt 4.5). Zum Ende des Kapitels in Abschnitt 4.7 soll auf mögliche Verbesserungen dieses Verfahrens eingegangen werden.

4.1 Motivation

Im Unterschied zu den zuvor betrachteten symbolischen Werten besteht bei den Analogwerten eine Art „Nachbarschaftsbeziehung“, das heißt bei Ähnlichkeit der Datenwerte sind diese auf die tatsächliche Eigenart der Daten zurückzuführen; einzelne Werte haben eine meßbare Beziehung zueinander im Gegensatz zu den symbolischen, zum Teil kodierten Werten. Dieses Faktum ermöglicht eine Klassifizierung mit Hilfe von künstlichen, neuronalen Netzen, wobei im Gegensatz zu den symbolischen Werten fehlertolerante, approximierende Zuordnungen der Datenwerte getroffen werden können. Eine solche Klassifizierung kann auf vielfältige Weise realisiert werden. Da die Daten inklusive der Klassenzuordnungen - in diesem Fall den Mißbrauchsarten - zur Verfügung stehen, ist ein überwachtes Lernen sinnvoll und möglich. Entsprechend Patterson ([Pat97] Seite 49 ff.) können verschiedene Netzwerktypen, wie zum Beispiel ADALINE, Backpropagation-Netze oder sogar „Self- Organizing Feature Maps“ und viele weitere, für diese Klassifizierung ausgewählt werden. Im Zuge dieser Diplomarbeit wurde das Modell der radialen Basisfunktionsnetze auf Grund der im Folgenden beschriebenen Eigenschaften vorgezogen. Dabei wurde das Training in Anlehnung an bekannte Backpropagationmethoden realisiert.

4.1.1 Aufgaben der Klassifizierung

Intention bei der Klassenzuordnung der Daten durch ein künstliches, neuronales Netz ist die automatische Spezialisierung auf bekannte Daten, beziehungsweise das selbstständige Training und Anlernen von vorgegebenen Mustern. Dabei ist besonders die fehlertolerante Abstraktion des Netzwerks gerade bei den zu untersuchenden analogen Daten von großer Bedeutung.

Ein neuronales Netz muß, bevor es zur Klassifizierung eingesetzt werden kann, trainiert werden. Aufgrund der Filterung durch die generalisierten Regeln (siehe Kapitel 3) stehen unter Umständen nur relativ wenige Datensätze zum Training und zur eventuell anschließenden Verifikation zur Verfügung[1]. Aus diesem Grund ist es notwendig, ein Netzmodell zu wählen, das mit möglichst wenigen Schritten schnell an das vorgegebene Szenario adaptiert.

Mit dem verwendeten Netz muß es möglich sein, verschiedene Konstellationen der Eingabedaten vorverarbeitend zu verrechnen, so daß auch Daten, wie zum Beispiel die Uhrzeit über die 24 Stundengrenze hinaus eine brauchbare „Nachbarschaftsbeziehung“ der Werte als Grundlage haben. Außerdem ist es für verschiedene Daten von Bedeutung, diese in Relation zu anderen Daten und Werten auszuwerten, da mit Hilfe dieser Daten allein, sonst nicht sinnvoll gearbeitet wer-

[1] Dies gilt vor allem für das in Abschnitt 6.2 vorgestellte Gesamtanalysemodell.

den kann. Als Beispiel diene das Geburtsdatum, das erst mit einer Verrechnung mit dem Transaktionsdatum zum Alter des Karteninhabers das Potential zu einer brauchbaren Analyse erhält.

Außerdem muß es möglich sein, mehrdimensionale Eingabevektoren unterschiedlicher Datentypen und Größenverhältnisse verarbeiten zu können. Eine Verarbeitung verschiedener Datentypen wie *Zeitwerte* oder *Geldbeträge* durch eine Netzschicht ist durch eine vorherige Normierung beziehungsweise Skalierung der Daten möglich. Dabei besteht jedoch unter Umständen durch eine unvorteilhafte Verrechnung der Daten die Gefahr, daß wichtige und aussagekräftige Daten nicht ausreichend Beachtung finden und damit nicht in ausreichendem Maße bei der Klassifizierung beteiligt werden. Eine andere Alternative ist die parallele Abarbeitung der einzelnen Datentypen in getrennten Unternetzwerken. Dadurch entfällt jedoch die Option, neue Zusammenhänge zwischen den einzelnen Analogdaten aufzudecken.

Ein weiterer nicht zu unterschätzender Punkt ist die Rückführbarkeit der Klassifikationsergebnisse. Bei vielen Netzmodellen ist es nicht möglich, die Klassenentscheidung nachzuvollziehen, sie arbeiten ähnlich einer „black box“. Dies ist jedoch bei dem vorliegenden Problem der Mißbrauchsprävention ein wichtiges Argument, erlaubt es doch, Rückschlüsse aus den Datenbeständen zu ziehen und folglich entsprechend präventiv reagieren zu können.

Zusammenfassend seien hier noch einmal die Kriterien für eine sinnvolle Netzauswahl aufgeführt:

1. Schneller Lernerfolg aufgrund weniger Daten; dadurch Zeitersparnis, was zu einer schnelleren Aktualisierung des Netzes führt.

2. Schnelle Abarbeitung, einfache Verrechnung der eintreffenden Daten; damit kurze Antwortzeit.

3. Möglichkeit der Verrechnung verschiedener Daten für eine sinnvolle Verarbeitung; Vorverarbeitung der Daten.

4. „Offene“ Datenverarbeitung, Möglichkeit der Rückführbarkeit der Klassenentscheidung und Klassenbestimmung.

4.1.2 Vergleich der Netztypen

Die oben genannten Bedingungen, die für die Klassifizierung der vorliegenden Daten von Bedeutung sind, sollen nun für die verschiedenen Netzmodelle diskutiert werden, um die Wahl bezüglich der radialen Basisfunktionsnetze argumentativ

zu begründen.

Ein Nachteil von Kohonens „Self Organizing Feature Maps“ besteht darin, daß während des Lernverfahrens ebenso wie beim normalen Lauf des Netzes globale Informationen notwendig sind. Das heißt bei jedem Schritt muß berechnet werden, welches der Neuronen im Neuronenverband die *größte* Aktivierung besitzt und demzufolge als aktiviert gilt [Roj93].
Im Gegensatz dazu werden bei den radialen Basisfunktionsnetzen zwar auch alle Aktivierungen berechnet, jedoch kommt es hier nicht allein auf eine maximale Aktivierung eines einzelnen Neurons an, die globalen Informationen müssen *nicht* in sortierter Reihenfolge vorliegen, also nicht untereinander verglichen werden. Radiale Basisfunktionsnetze sind in der Lage, durch die nicht-linearen Basisfunktionen im Kern des Netzes an den Grenzbereichen fließende Übergänge („unscharfe“ Zuordnungen) – ähnlich wie bei Fuzzy-Systemen – zu realisieren. Außerdem brauchen nur die Stützzellen betrachtet werden, die sich in der Nähe des Eingabemusters befinden. Nur diese Neuronen liefern mit der Aktivierungsfunktion von 0 verschiedene Werte. Damit handelt es sich um eine lokal agierende Netzarchitektur.
Für einfach strukturierte RBF-Netze ist es sogar möglich, direkte Berechnungen der Gewichte des Netzwerks durchzuführen, also diese nicht auf einem langwierigen, iterativen Wege zu bestimmen. Es ist damit unter Umständen möglich, RBF-Netze mit der direkten Gewichtsberechnung zu initialisieren und mit iterativen Verfahren wie Backpropagation nachzutrainieren [Zel97]. Mit dieser Art von Basisfunktionsnetzen ist es möglich, schnelle Lernerfolge zu erzielen und außerdem eine hohe Genauigkeit bei der Funktionsapproximation mit nur wenigen Neuronen zu erreichen. Je nach Bedarf kann in diesem Zusammenhang die Genauigkeit auf Kosten der Trainingsgeschwindigkeit durch Hinzunahme verschiedener, zu lernender Netzparameter komplexer gestaltet und optimiert werden. Diese Punkte ermöglichen somit kurze Trainingszeiten und eine schnelle Verarbeitung der zugrundeliegenden Daten.

Im Unterschied zur ADALINE besitzen radiale Basisfunktionsnetze jedoch nicht die Möglichkeit einer internen Verrechnung eingehender Datenwerte. Die Zusammenhänge der Eingabedaten können also nicht automatisch in Relation gesetzt und verrechnet werden. Jedoch bringt eine derartige Verrechnung wie sie bei der ADALINE möglich ist auch Nachteile mit sich. So ist eine nur langsame Konvergenzgeschwindigkeit der Gewichtungen des Netzes zu verzeichnen, sowie die Problematik gegeben, die unterschiedlich dimensionierten, verschiedenen Datenwerte wie Zeitwerte und Geldbeträge miteinander zu verrechnen. Deswegen ist es sinnvoll, die Daten vorher manuell in Relation zu setzen und so zum Beispiel Differenzen oder Quotienten zu bilden, die dann letztendlich von den Basisfunktionsnetzen verarbeitet werden. Diese Vorverarbeitung beziehungsweise Verrechnung der Eingabedaten kann unabhängig von dem eigentlichen Netztyp durchgeführt

werden.

Aus diesen aufgeführten Gründen wurde das Modell der radialen Basisfunktionsnetze für die Analyse der Analogdaten gewählt.

4.2 Methodik, Modellbeschreibung

Im Prinzip handelt es sich bei den radialen Basisfunktionsnetzen (RBF-Netzen) um mehrschichtige Approximationsnetzwerke ähnlich dem mehrschichtigen Perzeptron. Wird jedoch ein einfaches RBF-Netzwerk zu einer Mustererkennung beziehungsweise Klassifikation herangezogen, wird die eigentliche Problematik grundsätzlich durch eine nichtlineare Transformation oder Abbildung in einen hochdimensionalen Raum umgeformt [Hay94]. Der eigentliche Grund für eine solche Umwandlung basiert auf dem Theorem von Cover ([Cov65]) über Separierbarkeit von Mustern, in dem gezeigt wird, daß eine komplexe, nicht-lineare Musterklassifizierung in einem hochdimensionalen Raum einfacher linear zu separieren ist als in einem niedrigdimensionalem Raum. Liegen einmal linear separierbare Muster vor, so kann ein Perzeptron diese einfach ohne weiteres klassifizieren.

4.2.1 Netzarchitektur

Auf diesem Gedanken baut das Modell der RBF-Netze auf, wie sich im folgenden beschriebenen Aufbau widerspiegelt. Ein RBF-Netz besteht in der einfachen, ursprünglichen Form aus drei Schichten, wie auch in Abbildung 4.1 gezeigt. Ein solches Netz besteht aus einer Eingabeschicht, in der evtl. eine Vorverarbeitung der Daten stattfindet, einer verdeckten Schicht[2] aus radialen Basisfunktionsneuronen (RBF-Neuronen)(Ω) und einer linaren Ausgabeschicht bestehend aus beliebig vielen, einschichtigen Perzeptrons also einfachen linearen Neuronen (Σ).

4.2.2 Allgemeine Arbeitsweise von RBF- Netzen

Wie der Name dieser Art von Netzen aussagt, ist der wesentliche Kernpunkt der RBF-Netzwerke die radiale Basisfunktion (RBF). Bei diesen Basisfunktionen B_i handelt es sich um Glockenfunktionen[3], die jeweils nur lokal beschränkt den Eingaberaum abdecken und bewerten. Für die Basisfunktionen sind verschiedene Glockenfunktionen denkbar. Allgemein lassen sich diese Glockenfunktionen B_G

[2] *engl.* hidden Layer
[3] *engl.* bell shaped functions

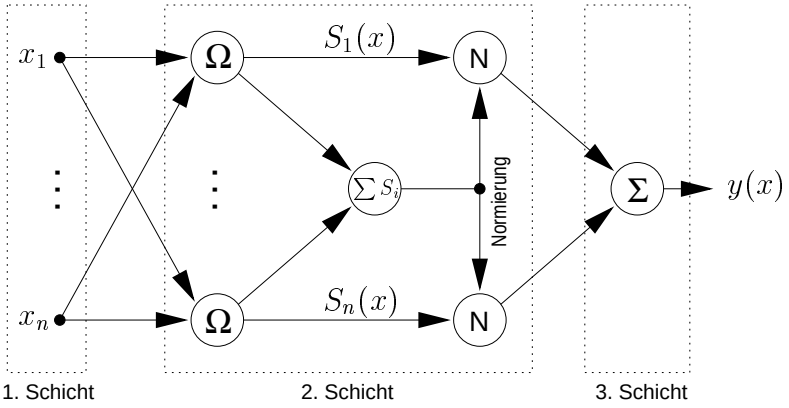


Abbildung 4.1: Ein Standard- RBF-Netzwerk in Anlehnung an [Bra95]

formal wie folgt definieren [Bra95]:

$$\begin{aligned} &B_G(-\infty) = B_G(\infty) = 0 \\ &\exists\ a \begin{cases} B_G(z) & \text{nicht anwachsend} \quad \forall z \in [a, \infty), \\ B_G(z) & \text{nicht abfallend} \quad \forall z \in (-\infty, a] \end{cases} \end{aligned} \tag{4.1}$$

Mit dieser Definition können diverse Glockenfunktionen konstruiert werden, sofern sie die Bedingung aus Gleichung 4.1 erfüllen. Für Glockenfunktionen nach obiger Definition kann man zeigen [CE92], daß von einem Netzwerk, das aus linear überlagerten, normierten Glockenfunktionen besteht, jede Funktion beliebig dicht approximiert werden kann[4]. Häufig wird die radiale Basisfunktion in Form der „Gaußschen Glockenfunktion“ mit

$$B_{Gi}(\mathbf{x}) = \exp\left(-0,5 \cdot \frac{(\mathbf{x} - \mathbf{c}_i)^2}{\sigma_i^2}\right) \tag{4.2}$$

verwendet. Damit sind die Einheiten der verborgenen Schicht, dem sogenannten Kern, jeweils durch das Zentrum $\mathbf{c}_i$ und der Glättungsfaktor σ_i gegeben. Der Glättungsfaktor σ_i bestimmt die Varianz, also die Ausdehnung des Einzugsgebietes rund um den Mittelpunkt $\mathbf{c}_i$ des i-ten RBF-Neurons. Die Ausgabe der einzelnen Einheiten ist eine nicht-lineare, radial symmetrische Funktion der Differenz aus Eingabe $\mathbf{x}_i$ und dem Zentrum $\mathbf{c}_i$. Somit ist die Ausgabe umso stärker je näher sich $\mathbf{x}_i$ an $\mathbf{c}_i$ befindet. Die RBFs sind Abbildungsfunktionen $B_G(\mathbf{x})$, wobei $B_G(\mathbf{x}) : \mathbb{R}^n \to \mathbb{R}$ gilt[Pat97].

Durch kleine Einzugsbereiche, die sich nicht überschneiden, kann keine Generalisierungsfähigkeit erreicht werden. Es ist daher sinnvoll, die Einzugsgebiete

[4]Als Beispiel können in diesem Zusammenhang die Graphiken in Abschnitt 4.6 dienen, in denen die Summe der verschiedenen Glockenfunktionen abgebildet ist.

sich überlappen zu lassen, so daß verschiedene Eingabevektoren Aktivität auf ein Neuron ausüben. Verwendet man also für die Varianz σ_i größere Werte, so erzielt man damit eine fehlertolerantere Klassifizierung. Dies ähnelt einem Assoziativspeicher, der den Eingaberaum adaptiv partitioniert und die Information der Trainingsdaten auf die Neurone in der unmittelbaren Umgebung verteilt.
Denkbar sind auch ellipsoidale Basisfunktionen, die sich besser an längsgerichtete Strukturen im Eingaberaum anpassen und in diesem Fall weniger RBF-Neuronen benötigt werden [Pie95]. Der Nachteil einer solchen Modellierung ist im größeren Berechnungs- und Lernaufwand für die zusätzlichen Parameter zu sehen.

Durch die lokale Beschränktheit der einzelnen RBF-Neurone, weisen die RBF-Netze in gewissem Maße eine Unabhängigkeit gegenüber der Reihenfolge der präsentierten Trainingsdaten auf. Dies unterstreicht die Möglichkeit des im nächsten Abschnitt näher spezifizierten „On-Line-Trainings“.

4.2.3 Training der RBF-Schicht

Zunächst muß zwischen zwei Arten des Lernens der RBF-Netze unterschieden werden:

On-Line-Lernen: Bei dieser Art des Lernens wird davon ausgegangen, daß die Trainingsdaten a priori nicht bekannt sind. In der Trainingsphase werden dann sukzessive die Musterdaten herangezogen, um das Netz zu trainieren.

Off-Line-Lernen: Hier sind sämtliche Trainingsdaten bekannt, und das Netz kann in jedem Trainingsschritt auf den kompletten Fundus der Daten zurückgreifen.

Initialisierung der verdeckten RBF-Schicht

Die Initialisierung des Lernvorgangs kann durch zwei unterschiedliche Arten, abhängig von den zu lernenden Daten, durchgeführt werden. Sind die Trainingsdaten bekannt, so ist eine effiziente Initialisierung der Neuronenzentren durch eine grobe Clusteranalyse der Eingabedaten möglich, ansonsten muß auf eine zufällige oder gleichmäßige Abdeckung des Musterraumes mit Neuronen zurückgegriffen werden. Dabei sind die radialen Basisfunktionen so zu wählen, daß sie zunächst einen großen Raum abdecken – die Varianz also hoch gewählt wird – und mit zunehmendem Lernstadium die Varianz verringert und damit der Einzugsbereich der einzelnen Neuronen verkleinert wird – die Neuronen sich also spezialisieren.
Eine andere Methode, den Eingaberaum mit den notwendigen RBF-Neuronen abzudecken, besteht darin, in Folge eines Online-Trainings die RBF-Neurone an die Stelle des Eingabevektors einzufügen, soweit kein adäquates, der gleichen Klasse zugeordnetes Neuron in der Nähe liegt. Während des fortschreitenden Trainings

können diese Neuronen dann den Rahmenbedingungen entsprechend Schritt für Schritt angepaßt werden.

Dynamisches Hinzufügen von Neuronen

In beiden Situationen, bei bekannten und unbekannten Trainingsdaten, lassen sich die Komplexität des Netzwerks und die Trainingszeiten deutlich verringern, wenn man das Netzwerk durch die sukzessive Erzeugung neuer Glockenfunktionen beziehungsweise neuer RBF-Neuronen verbessert. Um eine einigermaßen konsistente Überdeckung des Eingaberaumes zu erreichen, kann auf eine Fehlerschranke zurückgegriffen werden, mit der bestimmt wird, ab welcher Fehlergröße ein neues Neuron hinzugefügt werden darf. Diese Art von Strategie empfiehlt sich bei sequentieller Approximation unbekannter Daten [Bra95], ist also in Bezug zum „On-Line-Lernen“ zu sehen.
Die Ausdehnung und Lage der im Bereich des neuen Neurons angesiedelten Nachbarneurone kann je nach Klassenzuordnung zusätzlich variiert werden, um die verschiedenen Klassen gegeneinander schärfer abzugrenzen.
Die Wahl der Zentren $\mathbf{c}$ ist im allgemeinen nicht optimal. Wenn man diese Zentren verschiebt, kann die Leistung des RBF-Netzes im Verlauf des Trainings verbessert werden[Zel97].

Adaption durch Backpropagation

Das Backpropagation-Verfahren ist wohl das meistverwendete Lernverfahren für mehrschichtige, neuronale Netzwerke. Deswegen ist es ebenso möglich, dieses Verfahren in Verbindung mit RBF-Netzen einzusetzen wie in Zusammenhang mit Perzeptrons (z.B. ADALINE). Es kann wie auf Perceptrons ebenso auf die Trainingsalgorithmen der RBF-Netze angewendet werden, bei denen die verwendeten Parameter in den Trainingsschritten entsprechend den zugrundeliegenden Daten adaptiert werden. Im wesentlichen beruht dabei die Parameteranpassung auf einem Gradientenabstieg über einer Fehlerfunktion, die auf das festgelegte Fehlermaß zurückgreift. Als Fehlermaß wird meist der quadratische Fehler E verwendet.

$$E = \frac{1}{2} \sum_{m=1}^{M} \left(f(\mathbf{x}_m) - f^{soll}(\mathbf{x}_m) \right)^2 \tag{4.3}$$

Dabei ist $\mathbf{x}_m$ das m-te Eingabemuster und $f^{soll}(\mathbf{x}_m)$ die zu dem Eingabemuster gewünschte Ausgabe, der Sollwert. Dabei entspricht die Menge M der Menge aller Trainingsmuster. Bei der Auswahl des Trainingsmusterraumes ist zu beachten, daß es sich um eine repräsentative Stichprobe aus dem gesamten Eingaberaum handelt.

Die benötigten Gradienten für das Gradientenabstiegsverfahren sollen im folgenden angegeben werden. Zur einfacheren Darstellung gilt für die Berechnungen:

$$\Delta_m = f(\mathbf{x}_m) - f^{soll}(\mathbf{x}_m) \tag{4.4}$$

als Fehlermaß des Musters m. Damit ergibt sich für die partiellen Ableitungen von B_{G_i} für $\mathbf{c}_i$ und σ_i:

$$\begin{aligned} \frac{\delta B_{G_i}(\mathbf{x}_m)}{\delta \mathbf{c}_i} &= B_{G_i}(\mathbf{x}_m) \cdot \frac{\delta \frac{\|\mathbf{x}_m - \mathbf{c}_i\|^2}{\sigma^2}}{\delta \mathbf{c}_i} = -B_{G_i} \cdot 2 \cdot (\|\mathbf{x}_m - \mathbf{c}_i\|) \\ \frac{\delta B_{G_i}(\mathbf{x}_m)}{\delta \sigma_i} &= B_{G_i}(\mathbf{x}_m) \cdot \frac{\delta \frac{\|\mathbf{x}_m - \mathbf{c}_i\|^2}{\sigma^2}}{\delta \sigma_i} = -B_{G_i} \cdot \frac{2 \cdot (\|\mathbf{x}_m - \mathbf{c}_i\|)^2}{\sigma_i^3} \end{aligned} \tag{4.5}$$

Mit dem quadratischen Fehler E kann nun in Anlehnung an [Zel97] und [Pie95] der jeweilige Gradient berechnet und damit die Lernregeln beschrieben werden. Zunächst wird die Lernregel vorgestellt, mit der die Platzierung der Zentren mit Hilfe des Gradientenabstiegs realisiert werden kann:

$$\mathbf{c}_i(n+1) = \mathbf{c}_i(n) - \gamma \frac{\delta E(n)}{\delta \mathbf{c}_i(n)} \tag{4.6}$$

mit dem Gradienten

$$\frac{\delta E(n)}{\delta \mathbf{c}_i(n)} = \sum_{m=1}^{M} -\Delta_m \cdot w_i \cdot \frac{\delta B_{G_i}(\mathbf{x}_m)}{\delta \mathbf{c}_i} = -\sum_{m=1}^{M} \left(\Delta_m \cdot w_i \cdot \frac{\delta B_{G_i}(\mathbf{x}_m)}{\delta \mathbf{c}_i} \cdot B_{G_i} \right) \tag{4.7}$$

Für die Varianz beziehungsweise die Abdeckung der RBF-Neurone gilt entsprechend:

$$\sigma_i(n+1) = \sigma_i(n) - \gamma \frac{\delta E(n)}{\delta \sigma_i(n)} \tag{4.8}$$

mit dem Gradienten

$$\frac{\delta E(n)}{\delta \sigma_i(n)} = \sum_{m=1}^{M} -\Delta_m \cdot w_i \cdot \frac{\delta B_{G_i}(\mathbf{x}_m)}{\delta \sigma_i} = -\sum_{m=1}^{M} \left(\Delta_m \cdot w_i \cdot \frac{\delta B_{G_i}(\mathbf{x}_m)}{\delta \sigma_i} \cdot B_{G_i} \right) \tag{4.9}$$

Adaption durch selbstorganisierende Karten

Eine andere Variante der Adaption der Zentrumsparameter in der verdeckten Schicht lehnt an die Methode der **selbstorganisierenden Karten**[5] oder „Kohonen Karten“ an. Dabei wird die Nachbarschaftsbeziehung der einzelnen Neurone ausgenutzt und damit die Zentren im Laufe des Trainings zu einer homogenen Verteilung verschoben. Die Lernregel für diese Art der Adaption lautet:

[5] „Self Organizing Feature Maps“

$$\mathbf{c}_i(t+1) = \mathbf{c}_i(t) + \gamma(t+1)\,(\mathbf{x} - \mathbf{c}_i(t)) \tag{4.10}$$

wobei $\gamma(t+1)$ die aktuelle Lernrate im Lernschritt $t+1$ ist.

Entsprechend einer Methode von Xu, Krzyzak und Oja [XKO93], die sich an den Grundgedanken des „Wettbewerbs-Lernen“[6] anlehnt, kann bei den Trainingsiterationen das nächstliegende, klassengleiche Neuron dem aktuellen Trainingspunkt genähert werden. Entsprechend werden die nächstgelegenen, klassenfremden Neuronen von dem Trainingspunkt entfernt. Weiter können zusätzliche, weiter entfernte Neuronen (zweite Gewinner) vom Trainingspunkt distanziert werden, so daß die Cluster nach einiger Zeit *eindeutig* von einzelnen RBF- Neuronen beschrieben werden. Das bedeutet, daß anstelle der „winner-take-all“ Entscheidung eine „soft-winner-take-all“ Entscheidung verwendet wird, die die umliegenden, klassengleichen Neurone des Gewinnerneurons von dem Eingabevektor $\mathbf{x}$ zusätzlich zu der eigentlichen Zentrumsverschiebung distanziert.
Da sich die einzelnen RBF-Neurone nur auf lokale Bereiche rund um das Zentrum $\mathbf{c}_i$ beziehen, brauchen in der Trainingsphase jeweils nur die Neurone angepaßt werden, bei denen der Eingabevektor im unmittelbaren Einflußbereich liegt. Es findet also im Gegensatz zu Neuronen mit sigmoidaler Aktivierungsfunktion, die den Eingaberaum in zwei Halbräume aufteilen, ein lokal beschränktes Lernen auf Teilbereichen des Eingaberaumes statt. Durch dieses spezialisierte Training kann die Anpassung des Netzwerkes an die Trainingsdaten beschleunigt werden, da immer nur ein Neuron im Falle der „winner-take-all“-Lösung adaptiert werden muß.

Von [CE92] und [MD89] stammt ein Modell, das die Ausgabe der einzelnen RBFs normalisiert, wie es auch in der Abbildung 4.1, mit `N` bezeichnet, vorgesehen ist:

$$B_i(\mathbf{x}) = \frac{\frac{\exp(\mathbf{x}-\mathbf{w}_i^2)}{2\sigma_i^2}}{\sum_k \frac{\exp(\mathbf{x}-\mathbf{w}_k^2)}{2\sigma_k^2}} \tag{4.11}$$

Die Ausgabe des RBF-Netzes entspricht also einer Gewichtung der Durchschnittswerte der RBFs; nur *die* Gewichte sind entscheidend, deren zugehöriges Neuron in der verdeckten Schicht weitestgehend aktiviert ist, der Eingabevektor also in unmittelbarer Nähe des Zentrums liegt. Die übrigen Neurone zeigen kaum oder gar keine Aktivität und wirken sich somit nur wenig oder gar nicht auf die Gewichtung und die Ausgabe aus. Mit einem derartigen Netzwerk ist es möglich, jede Funktion beliebig dicht approximieren zu können.

[6] *engl.:* competitive learning

Anpassung der Gewichte der Ausgabeschicht

Die Ausgaben der verdeckten Schicht werden in der Ausgabeschicht des Netzwerkes, bestehend aus einem oder mehreren Perceptrons, weiterverarbeitet. Die Anzahl der Perceptrons hängt von der Dimensionalität des Ausgaberaums ab. Bei den Ausgabeneuronen handelt es sich um sogenannte „Sigmaneuronen“[7]. Diese erhalten jeweils als Eingabevektor die Ausgaben aller in der verdeckten Schicht befindlichen RBF-Neurone, den Vektor $\mathbf{x}$, und verrechnen diese mit den entsprechenden gelernten Gewichtsvektoren $\mathbf{w}_{kl}$. Für ein solches Sigmaneuron S_i gilt als Aktivitätsfunktion unter Zunahme eines Schwellwertes (Bias) s:

$$S_i(\mathbf{x}, \mathbf{w}) = \sum_j w_{ij} x_j - s = \mathbf{w}^T \mathbf{x} - s \tag{4.12}$$

Die Gewichte $\mathbf{w}$, die RBF-Neurone und Ausgabeschicht verbinden, werden mit Hilfe der Widrow-Hoff-Lernregel, die auch als „Delta-Lernregel“ bekannt ist, gelernt:

$$\mathbf{w}(t) = \mathbf{w}(t-1) - \gamma(t)\frac{\delta E}{\delta \mathbf{w}} = \mathbf{w}(t-1) - \gamma(t)\frac{(\mathbf{w}^T\mathbf{x} - L(\mathbf{x}))\mathbf{x}}{\mathbf{x}^2} \tag{4.13}$$

Wobei $\gamma(t)$ die veränderliche Lernrate zum Zeitpunkt t ist, und $L(\mathbf{x})$ die gewünschte Ausgabe, den *Sollwert*, für die Eingabe $\mathbf{x}$ darstellt. Der quadratische Fehler E berechnet sich dabei wie folgt:

$$E = \frac{1}{2}\left(\sum_i^n \left(\mathbf{x} \cdot w_i - L(\mathbf{x})\right)\right)^2 \tag{4.14}$$

Die Ausgabe dieser letzten Schicht kann nun entweder bei Angabe einer Schwelle von binärer oder sonst analoger Natur sein.

4.3 Aufbau eines RBF- Netzes

Im wesentlichen wurde die Netzarchitektur an bekannte RBF-Netzmodelle angelehnt (siehe [Bra95] Seite 308 ff.). Die Möglichkeit der objekt-orientierten Entwicklung und Implementierung gibt eine Einteilung in verschiedene Klassen fast von selbst vor. So konnte das Netz in die einzelnen Schichten, Neurone und Gewichte klassenspezifisch zerlegt und nahezu direkt implementiert werden. Eine genauere Aufstellung der Klassen ist im Anhang D aufgeführt. Es sei an dieser Stelle außerdem auf die beiliegende HTML-Dokumentation auf der CD-Rom verwiesen, die eine genaue Beschreibung der Klassen und Methoden enthält.

[7] *engl.* sigma units

4.3.1 Allgemeine Grundlagen

Zu den vom Benutzer zu definierenden Parametern zählen für jedes RBF-Netz:

Lernrate: Bestimmt die initiale Lernrate γ des einzelnen Netzes. Je nach Lernstadium können hier unterschiedliche Werte eingetragen werden.

Initiales Gewicht: Legt den Wert der initialen Gewichte fest, die die RBF-Neuronen und die Ausgabeschicht verbinden. Da die Gewichte selbstständig gelernt werden, erübrigt es sich gewöhnlich hier einen von 0 verschiedenen Wert einzutragen.

Initiale Varianz: Legt die Abdeckung eines neuen RBF-Neurons fest. Es wird damit die Ausdehnung der radialen Basisfunktion festgesetzt, die zu Beginn den Einfluß des Neurons kontrolliert. Je nach Datumswert kann dieser Wert zwischen 0,5 und mehreren 1000 variieren.

Die Arbeit des Netzwerkes läßt sich in zwei verschiedene Phasen einteilen, die Trainingsphase und die Arbeitsphase. Die Arbeitsphase unterscheidet sich von der Trainingsphase im wesentlichen nur dadurch, daß sämtliche Parameter beibehalten und keine Änderungen mehr an diesen vorgenommen werden. Diese Phase wird zur Überprüfung der Klassifizierungsqualität des Netzes verwendet.
Der Lernalgorithmus wurde als „On-Line“-Algorithmus implementiert. Es ist also nicht vorgesehen, die Trainingsmustermenge beliebig oft zu durchlaufen, sondern ausschließlich das Netz auf Basis immer wieder neuer Daten zu trainieren. Diese Methode ist an die Umstände der Realität angepaßt, da ein ständiges Training des Netzes, parallel zur Klassifizierung, dieses auf einem möglichst aktuellen Stand hält.
In der Ausgabeschicht befindet sich ein einzelnes Sigmaneuron. Mit diesem Neuron ist eine Schwellenwertfunktion verbunden, die eine binäre Entscheidung ermöglicht. Dies ist insofern ausreichend, als eine binäre Klassenentscheidung gefordert ist, um Mißbrauch und legale Transaktion zu unterscheiden. Eine differenziertere Ausgabe beziehungsweise Unterteilung in die verschiedenen Mißbrauchsklassen kann unter Umständen besser mit einer größeren Anzahl an Ausgabeeinheiten realisiert werden.
Die Datenrekrutierung findet unter anderem entsprechend den Einschränkungen durch die generalisierten Regeln direkt in der Datenbank mit sämtlichen Transaktions- und Karteninhaberdaten statt. Dabei können die einzelnen Datensätze aus den einzelnen Datenbereichen sukzessive – also legale, beziehungsweise illegale Transaktionen – oder auch zufällig ausgewählt werden. Außerdem ist es auf diese Weise einfach möglich, weitere Einschränkungen in Form von SQL-Abfragen und Bedingungen zu formulieren und somit Einfluß auf die zu verarbeitenden Daten zu haben, und diese steuern, beziehungsweise überwachen zu können.

4.3.2 Training

In der Trainingsphase werden die Parameter der verdeckten Schicht, Lage $\mathbf{c}_i$ und Ausdehnung σ_i der enthaltenen RBF-Neurone, sowie die Gewichte w_i der Ausgabeschicht an die Trainingsdaten angepaßt. Im Laufe des Trainings verringert sich die initiale Lernrate mit jedem Iterationsschritt, um so eine langsame Festigung der Werte zu erzielen.

Dynamisches Hinzufügen von Neuronen

In der Trainingsphase wird zunächst von einem Netzwerk **ohne** RBF-Neurone in der verdeckten Schicht ausgegangen. Diese werden im Laufe des Trainings bei Bedarf entsprechend den Eingabevektoren eingefügt. Bedarf besteht, sobald in der unmittelbaren Umgebung des Eingabevektors sich *kein* Neuron der gleichen Klasse oder ausschließlich Neurone mit einer anderen Klassenzuordnung befinden. In dem Programmausschnitt 4.1 auf Seite 78 wird zum Beispiel ein Neuron in den Zeilen 9 und 15 eingefügt, sowie am Anfang, wenn noch gar kein Neuron in der verdeckten Schicht enthalten ist (Zeile 26). Auf diese Weise können Werte, die an außergewöhnlichen Punkten eine Klasse vertreten und außerhalb üblicher Bereiche liegen, dem Netz antrainiert und für die spätere Klassifizierung gespeichert werden. Ziel ist es ja gerade auch, außergewöhnliche Datenwerte, beziehungsweise Datenkombinationen aufzudecken und auch zuordnen zu können.

Um eine möglichst konsistente Neuronenverteilung in der verdeckten Schicht zu erzielen, wurde eine Schwelle eingeführt, die bestimmt, unter welchen Bedingungen ein neues Neuron eingefügt werden darf. Diese Schwelle ist abhängig von der Varianz und der Klassenzuordnung der umliegenden Neurone. So wird ein Neuron eingefügt, wenn einer der folgend aufgeführten Punkte zutrifft:

- (Distanz zum nächsten Neuron $>$ Varianz)
  $\wedge$ (Klassenzuordnung des nächsten Neurons $\neq$ Sollwert)

- (Distanz zum nächsten Neuron $> 2 \cdot$ Varianz)
  $\wedge$ (Klassenzuordnung des nächsten Neurons $=$ Sollwert)

Damit nicht mit fortgeschrittenem Lernprozeß unnötig viele Neurone verwaltet werden müssen, ist es denkbar, eine „Lebenszeit“ für die Neurone einzuführen. Werden die Neurone innerhalb dieser Zeitintervalle nicht aktiviert, es wird keine Ausgabe größer 0 von diesem Neuron registriert, so kann das Neuron aus der verdeckten Schicht, wie auch das zugehörige Gewicht in der Ausgabeschicht, entfernt werden.

Diese Methode baut außerdem darauf auf, daß hauptsächlich legale Transaktionen, die den Normalfall darstellen, verarbeitet werden. Diese Wertemuster können

mit nur wenigen Neuronen ausreichend abgedeckt werden. Für den Fall der Mißbräuche werden eher neue Neuronen angelegt, da es sich dabei hauptsächlich um eben diese ungewöhnlichen Eingabedaten handelt, für die ein neues Neuron angelegt wird. Dies äußert sich im Verhältnis aus Trainingsdaten und Klassenzuordnungen der Neuronen. Es konnte häufig folgende Relation festgestellt werden:

$$\frac{\#\text{ legale Trainingsdaten}}{\#\text{ Neuronen (Klasse } \textit{legal})} < \frac{\#\text{ illegale Trainingsdaten}}{\#\text{ Neuronen (Klasse } \textit{illegal})} \tag{4.15}$$

wie aus Tabelle 4.1 zu entnehmen ist. Diese Ergebnissen wurden bei einem Trainingslauf mit 200 Transaktionen im Verhältnis 1:10 ermittelt[8]. Es sind jeweils die Absolutzahlen (Spalte 2 und 4) an Neuronen, wie auch der jeweilige prozentuale Anteil (Spalte 3 und 5) an Neuronen zu den entsprechend trainierten Transaktionen (Zeile 2) angegeben.

Beschreibung	Zuordnung			
	legale	%	illegale	%
Gesamttransaktionen	182		18	
Beträge	36	19,8	17	94,4
Betrag + Aval. Balance	57	31,3	8	44,4
Betrag + Kreditlimit	47	25,8	3	16,6
Karteninhaberalter	23	12,6	9	50,0
Ablaufzeitspanne	15	8,24	4	22,2
Öffnungszeitspanne	23	12,6	8	44,4
Uhrzeit	22	12,1	3	16,6

Tabelle 4.1: Neuronenverhältnisse

Neuronenpositionierung und Zentrumsadaption

Nachdem nun verschiedene Neuronen angelegt sind, kann es passieren, daß ein Eingabevektor in die Nähe eines der vorhandenen Neurone gelangt. In diesem Fall wird entsprechend der Klassenzuordnung und dem Abstand von dem nächstgelegenen Neuron N_{next} reagiert (Vergleiche Programmkode 4.1). Zunächst wird unterschieden, ob der Eingabevektor in dem Intervall $I_{N_{next}} = (\sigma_{N_{next}}, 2\sigma_{N_{next}})$ liegt (Zeile 10). Ist dies der Fall, und stimmt die Klassenzuordnung ebenfalls überein, so wird das nächste Neuron N_{next} in Richtung des Eingabevektors entsprechend der Lernrate verschoben (Gleichung 4.16 und Zeile 11).

$$\mathbf{c}_{next}(t+1) = \mathbf{c}_{next}(t) + \gamma(t) \cdot (\mathbf{c}_{next}(t) - \mathbf{x}) \tag{4.16}$$

[8]Auf jeden zum Training verwendeten illegalen Datensatz kommen 10 legale Datensätze.

Stimmt jedoch die Klassenzuordnung der Eingabe und des nächsten Neurons **nicht** überein, so wird das Zentrum des Neurons N_{next} in die entgegengesetzte Richtung verschoben (Gleichung 4.17 und Zeile 22).

$$\mathbf{c}_{next}(t+1) = \mathbf{c}_{next}(t) - \gamma(t) \cdot (\mathbf{c}_{next}(t) - \mathbf{x}) \tag{4.17}$$

Dies äußert sich an dem negativen Vorzeichen der Lernrate γ in Gleichung 4.17.

Varianzanpassung

Entsprechend der Positionierung der RBF-Neuronen wird auch deren Abdeckung beziehungsweise deren Ausdehnung im Laufe des Trainingsvorgangs verändert. Dies kann auf zweierlei Arten passieren. Zum einen kann es zu einer Verallgemeinerung des Neurons N_{next} kommen, die Varianz $\sigma_{N_{next}}$ wird vergrößert. Im anderen Fall einer Spezialisierung kommt es zu einer Verkleinerung der Varianz $\sigma_{N_{next}}$, indem die Lernrate negativ in die Verrechnung eingeht. Erster Fall tritt ein, wenn gilt: $\|\mathbf{c}_{N_{next}} - \mathbf{x}\| \in (0, 2 \cdot \sigma_{N_{next}})$ und die Klassenzuordnung von Neuron N_{next} dem Sollwert vom Eingabevektor $\mathbf{x}$ entspricht (Zeile 12).
Der zweite Fall hingegen tritt auf, wenn die Klassenzuordnung **nicht** dem Sollwert des Eingabevektors $\mathbf{x}$ entspricht und außerdem gilt: $\|\mathbf{c}_{N_{next}} - \mathbf{x}\| < \sigma_{N_{next}}$ (Zeile 23). Die Varianz berechnet sich dann wie folgt:

$$\sigma_{N_{next}}(t+1) = \sigma_{N_{next}}(t) \cdot \gamma(t) \tag{4.18}$$

beziehungsweise

$$\sigma_{N_{next}}(t+1) = \sigma_{N_{next}}(t) \cdot -\gamma(t) \tag{4.19}$$

Algorithmus 4.1

```
1  if (neuronVec.size() ≥ 1)                                  schon Neuronen vorhanden?
2    then
3        if (minDist > width)                                 Wert außerhalb Neuron-Einzugsbereichs?
4          then
5              if (desired = N_next.getNeuroClass())          Klassenzuordnung gleich?
6                then
7                    if (minDist > 2 * width)                sehr weit entfernt?
8                      then
9                          neuronVec.addNeuron();            füge Neuron hinzu
10                       else                                nicht weit entfernt!
11                         N_next.moveCentre(inVector, learnrate);   „hin“
12                         N_next.raiseWidth(learnrate);     vergrößere Varianz
13                   fi
14               else
15                   neuronVec.addNeuron();                  füge Neuron hinzu
16             fi
17         else                                              nicht sehr weit entfernt
18             if (desired = N_next.getNeuroClass())         Klassenzuordnung gleich?
19               then
20                   N_next.raiseWidth(learnrate);           vergrößere Varianz
21               else                                        Klassenzuordung verschieden!
22                   N_next.moveCentre(inVector, (−1 * learnrate));   „weg“
23                   N_next.raiseWidth((−1 * learnrate));    vermindere Varianz
24             fi
25         else                                              kein nächstes Neuron
26             neuronVec.addNeuron();                        erstes Neuron wird angelegt!
27       fi
28 fi
```

Methoden	Beschreibung
addNeuron()	fügt Neuron hinzu
moveCentre()	verschiebt Neuron
raiseWidth()	verändert Varianz
getNeuroClass()	gibt Klassenzugehörigkeit des Neurons zurück
size()	bestimmt Größe des Neuronenvektors der verdeckten Schicht

Methoden

Methoden	Beschreibung
inVector	Eingabevektor
neuronVec	Neuronenvektor
N_{next}	nächst gelegenes Neuron
minDist	minimale Distanz zum nächsten Neuron
desired	Sollwert des Eingabevektors
width	Varianz
learnrate	Lernrate

Variablen, Parameter

Tabelle 4.2: Erläuterungen zum Algorithmus 4.1

Training der Ausgabeschicht

Die Ausgabeschicht bestehend aus einem linearen Neuron wird entsprechend der Lernregel 4.13 auf Seite 73 trainiert. Die binäre Ausgabe kann dabei durch die Heavyside-Funktion oder Vorzeichenfunktion (sgn(.)) entsprechend Gleichung 4.20 gegeben (siehe auch Abbildung 4.2) sein:

$$f(y) = sgn(y) = \begin{cases} +1 & z \geq 0 \\ -1 & z < 0 \end{cases} \tag{4.20}$$

Die Binärfunktionen unterscheiden sich prinzipiell nicht, so kann man durch einfache Verrechnung die gewählte Funktionen ineinander überführen. Trotzdem bietet die Vorzeichenfunktion den Vorteil, daß der Wert 0 umgangen wird, der für eine Verrechnung problematisch sein kann. Für die implementierten Netze wurde die

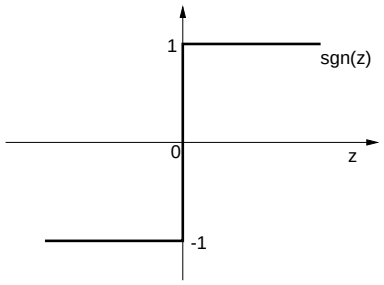


Abbildung 4.2: Vorzeichenfunktion (sgn(.)) nach [Bra95]

Vorzeichenfunktion (sgn(.)) gewählt. Es wurden die Ausgabewerte $+1$ für eine Klassifizierung in die Menge der legalen Daten gewählt und der Wert -1 für eine Einordnung in die Mißbrauchsklasse.

4.4 Klassifizierung auf den Transaktionsdaten

4.4.1 Multinetzarchitektur

Aufgrund der bei gewöhnlichen RBF-Netzen fehlenden Schicht zur Verrechnung der Eingabedaten (siehe auch Abschnitt 4.1.1) muß eine weitere Schicht implementiert werden, die eine solche Vorverarbeitung bewerkstelligt. Erst eine derartige Aufbereitung der Daten macht es möglich, viele der vorliegenden Daten auf eine sinnvolle Weise verarbeiten zu können. Datentypen wie Geburtsdatum (`BIRTH_DT`) oder Transaktionsbetrag (`TRN_AMNT`) können nur bedingt ohne weitere Behandlung zu einer aussagekräftigen Klassifizierung beitragen, denn bei ersterem handelt es sich um einen stetig wachsenden Wert, letzterer kann nur in Ausnahmesituationen spezifisch für eine Zuordnungsklasse sein. Aus diesem Grunde ist es notwendig, diese Werte in Relation zu anderen Werten zu setzen.

So erhält man aus dem Geburtsdatum in Zusammenhang mit dem Transaktionsdatum das Alter des Karteninhabers zum Zeitpunkt der Transaktion, oder im Falle des Transaktionsbetrags in Verbindung mit dem Kreditlimit eine Angabe, wie stark das Konto ausgelastet ist, beziehungsweise wie stark es belastet wird. Um die Verrechnung der Daten in benutzerdefinierte Bahnen zu lenken, wurden verschiedene Eingabedaten zusammengefaßt, verrechnet und einer Netzinstanz der oben beschriebenen RBF-Netze zur Verarbeitung als Eingabevektor gegeben. Die auf diese Weise gewonnenen spezialisierten Unter-RBF-Netzwerke werden seriell – der Reihe nach – ausgewertet. Die Ausgaben der einzelnen Netze werden in einem abschließenden linearen Neuron (*Sigmaneuron*) zusammengefaßt und gewichtet und zu einer binären Ausgabe entsprechend dem Ausgabeneuron der beschriebenen RBF-Netze verarbeitet. In Abbildung 4.3 ist der Aufbau der Netzwerkarchitektur zur Klassifizierung dargestellt.
Ein weiterer Punkt in diesem Zusammenhang ist das Problem der unterschiedlichen Distanzmaße, die bei den verschiedenen Datentypen auftreten, jedoch mit der vorliegenden Netzarchitektur umgangen werden können. Jedes RBF-Unternetz verarbeitet somit einen speziellen Typus an Daten, wie zum Beispiel Datumsangaben, Zeitdifferenzen oder Finanzbeträge. Es wurden sieben, im folgenden näher beschriebene Unternetze, beziehungsweise vorverarbeitende Filter implementiert. Diese Netzarchitektur kann entsprechend den Bedürfnissen und Anforderungen durch Hinzunahme, Abwandlung oder Sperrung verschiedener spezialisierter Unternetztypen angepaßt werden.

Desweiteren ist in diesem Zusammenhang die Möglichkeit gegeben, rückführend auf den Kern der einzelnen, spezialisierten Unternetze zurückzugreifen, und somit die „Entscheidungsfindung“, beziehungsweise die Grundlage der Klassenentscheidung bezüglich der jeweils bearbeiteten Daten zurückzuverfolgen, und Rückschlüsse auf die Trainingsdatenbasis ziehen zu können.

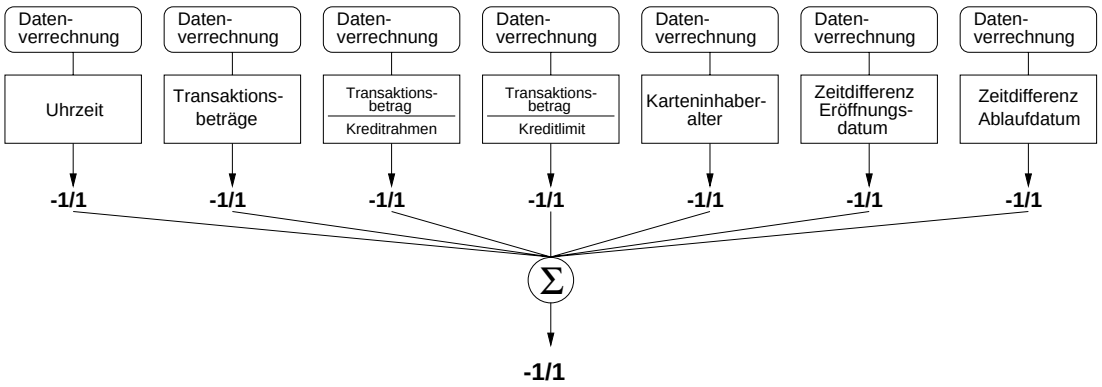


Abbildung 4.3: Netzarchitektur zur Klassifizierung der analogen Daten

4.4.2 Datenvorverarbeitung

Im folgenden werden die verschiedenen vorverarbeitenden Filter, die den einzelnen übereinstimmenden Standardnetzen vorgeschaltet sind, näher erläutert, sowie ihre Arbeitsweise beschrieben.
Eine Ausnahme stellt das die Transaktionsuhrzeit verarbeitende Netz dar, da es sich bei diesen Datenwerten um eine nicht-lineare Funktion handelt. Es ist zu berücksichtigen, daß auch Werte über den Tageswechsel um 24:00 Uhr hinaus eine Nachbarschaft besitzen. Dies spielt zwar vorrangig eine Rolle bei der netzinternen Distanzberechnung, soll jedoch trotzdem an dieser Stelle erläutert werden, da es sich in den Rahmen der speziellen Datenverarbeitung und -aufbereitung einfügt.

Zeitnetz: Dieses Netz benötigt zur Verarbeitung die Uhrzeit der Transaktion. Diese wird mit einer Transaktion übermittelt und liegt im Feld `TRN_DT` samt Datum vor. Der entsprechende Filter extrahiert die Uhrzeit aus dem Datum und wandelt es in einen Gleitkommawert um. Dieser kann problemlos als Eingabe für das darauffolgende RBF-Unternetz dienen.

Wie oben erwähnt, bedarf es bei der Klassifizierung der Uhrzeit einer Abwandlung in der Differenz- beziehungsweise Distanzberechnung. Diese wird wie folgt erreicht.

1. Zunächst wird die Differenz aus den relevanten Uhrzeiten gebildet. Als Beispiel diene das Uhrzeitpaar 23 Uhr und 3 Uhr in dieser Reihenfolge, das korrekterweise eine Differenz von 4 Stunden aufweist. Die Differenz, die sich berechnet, beträgt jedoch 20 Stunden.
2. Aus diesem Grunde wird nun, falls wie im vorliegenden Fall die Differenz mehr als 12 Stunden beträgt, diese von 24 Stunden abgezogen. Am Beispiel ergibt sich 24 Stunden - 20 Stunden = 4 Stunden, was der korrekten Zeitdifferenz entspricht.

Diese Abwandlung der Zeitdiffenzbildung ist bei der Implementierung der RBF-Netze berücksichtigt worden und wird bei Bedarf verwendet.

Beträge: Der Filter für die Geldbeträge einer Transaktion wandelt primär nur die Zahlenwerte in ein zur Verrechnung benötigtes Datenformat um. Jedoch dient der Filter auch als Puffer, der solange Werte aufnimmt, bis das eigentliche Netz mit dem nächsten Schritt ausgewertet wird. Letztendlich handelt es sich bei der Eingabe des Netzes also um einen drei-dimensionalen Vektor.

Zeitdifferenz zum Eröffnungsdatum: Dieser Filter berechnet zur anschließenden Netzauswertung die Zeitdifferenz aus Transaktionsdatum (`TRN_DT`)

und dem Eröffnungsdatum der Karte (START_DT) in Tagen, in dieser Reihenfolge. Karten, die vor dem Eröffnungstermin benutzt werden, erzeugen einen negativen Wert, der sich entsprechend in der anschließenden Netzverarbeitung absetzen wird oder sofort als Mißbrauch erkannt werden kann.

Zeitdifferenz zum Auslaufdatum: Wie das vorhergehende Netz wird hier die Zeitdifferenz aus dem Verfallsdatum der Karte(EXP_DT) und dem Transaktionsdatum (TRN_DT) berechnet. Auch hier gilt: Liegt das Transaktionsdatum hinter dem Verfallsdatum, so wird ein negativer Wert dem Netz zur Verarbeitung geliefert. In diesem Fall kann sofort auf Mißbrauch entschieden werden.

Alter: Beim Alter handelt es sich im Gegensatz zum eigentlich angegebenen Geburtsdatum um einen relativen Wert, der für eine derartige Verarbeitung notwendig ist. Das Alter wird in Jahren angegeben. Dazu wird das Transaktionsdatum mit in die Berechnung einbezogen. Die Differenz ergibt das Alter des Karteninhabers zum Zeitpunkt der Transaktion.

Kreditlimit und Transaktionsbetrag: Diese Art von Filter ist an den Betragsfilter angelehnt. Auch hier werden zunächst die Werte umgewandelt und zwischengespeichert, bis es zu einer Verrechnung aus Betragswert (TRN_AMT) und Kreditlimit (CR_LMT) zum Eingabevektor $\mathbf{x}$ kommt. Die beiden Werte werden wie folgt in Relation zueinander gestellt:

$$\mathbf{x} = \frac{\texttt{CR\_LMT}}{\texttt{TRN\_AMT}} \tag{4.21}$$

Verfügungsrahmen und Transaktionsbetrag: Dieser Filter arbeitet wie der vorhergegangene, indem er den Betragswert (TRN_AMT) und den Verfügungsrahmen (AVAL_BALLNCE) zueinander in Relation stellt. Das berechnete Verhältnis ist der Eingabevektor $\mathbf{x}$ dieses Unternetzes:

$$\mathbf{x} = \frac{\texttt{AVAL\_BALLNCE}}{\texttt{TRN\_AMT}} \tag{4.22}$$

Wie beschrieben, können die zusammengefaßten RBF-Netze als eine Art Netzschicht zur Klassifizierung der analogen Daten betrachtet werden. Die Ergebnisse der einzelnen RBF- Unternetze werden in einem einfachen, linearen Neuron zusammengefaßt und gewichtet. Die Ausgabe dieses eigentlich 4-schichtigen Netzwerks ist eine einfache binäre Klassenentscheidung in Mißbrauch oder nicht Mißbrauch.

An dieser Stelle sei nocheinmal der Ablauf der Klassifizierung in den wesentlichen Punkten zusammengefaßt:

1. Vorverarbeitung der unterschiedlichen Daten durch die oben beschriebenen Filter.

2. Verarbeitung der zum Teil verrechneten Eingabevektoren in der nicht-linearen, verdeckten Schicht durch die RBF-Neurone der einzelnen Netze.

3. Lineare Separierung, beziehungsweise Gewichtung der Ausgaben der verschiedenen verdeckten Schichten durch jeweils ein Sigmaneuron und anschließende Klassifizierung durch Schwellenwertoperation.

4. Gewichtung der binären Ausgaben aus den verschiedenen Unternetzen durch ein einzelnes lineares Neuron und Ausgabe einer ebenfalls binären Klassenentscheidung.

4.5 Ergebnisse der Klassifizierung

In diesem Abschnitt nun werden speziell die Ergebnisse der Auswertung der analogen Daten auf Basis der radialen Basisfunktionsnetze vorgestellt. In diesem Zusammenhang sind verschiedene Ansätze denkbar. Diese basieren unter anderem auf der Tatsache, daß das Training für die spätere Klassifizierung von ausschlaggebender Bedeutung sein kann. So können verschiedene Trainingsszenarien zu unterschiedlichen Klassifizierungsergebnissen führen. Diese Trainingsmethoden sollen hier vorgestellt und anhand der Ergebnisse ausgewertet und diskutiert werden. Weiter kann gezeigt werden, daß eine ausschließliche Einordnung der Transaktionen mit Hilfe der Klassifizierung durch die Analogdaten mit Hilfe der RBF-Netze allein keine Lösung zur Mißbrauchsprävention darstellt.

4.5.1 Training

Das Training ist für die spätere Klassifizierung von entscheidender Bedeutung. Verschiedene Trainigszenarien und Parameter, wie Lernrate, Initialgewicht oder -varianz, spielen dabei eine wesentliche Rolle und zeigen Auswirkungen auf die anschließende Klassifizierung in der Verifikationsphase. Außerdem wirkt sich ebenfalls das Klassenverhältnis der zum Training präsentierten Daten auf das Lern- und Klassifizierungsverhalten des Netzes aus. So ist zum Beispiel denkbar, durch ein ungleichmäßiges Training, das heißt vermehrt Datensätze einer Klasse dem Netz zu präsentieren, eine Zielrichtung bei der Klassenentscheidung vorzugeben. Doch zunächst soll auf die einzelnen Trainingsparameter eingegangen werden: Dazu sind in Tabelle 4.3 die in den einzelnen Unternetzwerken verwendeten Parameter aufgeführt. Diese wurden entsprechend der auszuwertenden Daten gewählt, beziehungsweise empirisch ermittelt.
Dabei kann mit dem Parameter der Initialvarianz, der Ausdehnung der radialen Basisfunktion, beim Anlegen eines neuen Neurons ein Maß für den Einzugsbe-

Unternetz	Lernrate	Initialgewicht	Initialvarianz
Uhrzeit	0,001	0	3
Betragswerte	0,05	0	100000
Eröffnungsdatum	0,0001	0	100
Auslaufdatum	0,0001	0	100
Karteninhaberalter	0,001	0	5
Kreditlimit	0,01	0	50
Verfügungsrahmen	0,01	0	50

Tabelle 4.3: Netzparameter der einzelnen Unternetzwerke

reich des neuen Neurons und somit für die Abstraktionfähigkeit des Neurons, beziehungsweise des Netzes angegeben werden. Je nach Fluktuation der entsprechenden, zu verarbeitenden Werte ist auch die Lernrate zu wählen. Handelt es sich um eher statische, beziehungsweise um nur wenig schwankende Werte, so kann mit einer kleinen Lernrate begonnen werden. Andernfalls sollte die Lernrate größer gewählt werden, um den unter Umständen erheblichen Wertschwankungen entgegen zuwirken
Weiter kann beim Training die Anzahl der Trainingszyklen variiert werden, um so ein bestmögliches Training für die anschließende Klassifizierung zu erreichen. Dabei ist zu beachten, daß es nicht zu einer Überanpassung[9] an die Trainingsdaten kommt. Dies hat zur Folge, daß aufgrund eines exzessiven Trainingprozesses Datenmuster „gemerkt“ werden, anstatt eine gute Verallgemeinerung bezüglich des Datenraumes zu erzielen. Dies kann unter Umständen durch eine sehr große, extensiv eingesetzte Traingsmenge hervorgerufen werden [Pat97]. Als Folge kann es zu einer Abnahme der Diagnoseleistung im Laufe des fortgesetzten Trainings, beziehungsweise während der eigentlichen Klassifizierung kommen.
Aus diesem Grunde wurden die Trefferquoten auch beim Training protokolliert und in Tabelle 4.4 aufgeführt.

Trainings-	Trefferquoten							Trefferquote
zyklen	A	B	C	D	E	F	G	Gesamtnetz
100	95,0	71,0	51,0	54,0	53,0	69,0	58,0	90,0
200	70,0	82,0	57,5	68,5	55,5	62,5	55,5	82,0
300	64,0	84,4	57,6	69,3	60,3	59,3	56,6	84,3
400	52,5	81,8	52,5	61,5	56,5	65,3	49,8	81,0
800	42,8	79,6	55,4	65,9	54,4	66,4	52,5	79,6

Tabelle 4.4: Trainingsergebnisse (Angaben in %)

Die alphabetischen Markierungen wurden stellvertretend für die einzelnen Unternetze benutzt; es gilt folgende Abbildung:

[9] *engl.*: overtraining

Bezeichnung	Unternetzart
A	tageszeitliche Auswertung (Transaktionsuhrzeit)
B	Transaktionsbetrag, Kreditlimit, Verfügungsrahmen
C	Zeitdifferenz zwischen Eröffnungs- und Transaktionsdatum
D	Zeitdifferenz zwischen Ablauf- und Transaktionsdatum
E	Karteninhaberalter
F	Buchungsbetrag in Relation zum Kreditlimit
G	Buchungsbetrag in Relation zum Verfügungsrahmen

Tabelle 4.5: Erläuterung zu Tabelle 4.4

4.5.2 Verifikation

In Tabelle 4.6 sind die Trefferquoten der anschließenden Verifikationsphase auf Basis von 500 Datensätzen, also jeweils 250 illegalen und legalen Datensätzen, aufgeführt. Hinzu kommen die einzelnen Trefferquoten für die legalen sowie illegalen Zuweisungen beziehungsweise die entsprechenden Fehlerquoten.

Trainings-zyklen	Treffer-quote	richtige Zuweisungen		falsche Zuweisungen		Konfidenz
		legale	illegale	legale	illegale	
100	46,4	4,8	88,0	95,2	12,0	0,092
200	74,8	76,4	73,2	23,6	26,8	0,309
300	82,2	86,0	78,4	14,0	21,6	0,557
400	69,6	94,8	44,4	5,2	55,6	0,847
800	83,0	89,2	76,8	10,8	23,2	0,706

Tabelle 4.6: Trefferquoten und Konfidenzwerte der Verifikationsphase (Angaben in %)

Ebenfalls sind in Tabelle 4.6 die Konfidenzwerte aufgeführt, die durch die ausschließliche Klassifizierung der Daten erzielt wurden. Dabei muß berücksichtigt werden, daß der Fehlalarmanteil auf ein reales Verhältnismaß hochgerechnet werden muß. Da ein Mißbrauchsanteil von 0,1% vorliegt, muß bei paritätischer Auswertung der Regeln also ein Hochrechnungsfaktor von 1000 angewandt werden. In Anlehnung an Gleichung 3.17 wird die Konfidenz also folgendermaßen berechnet:

$$confidence = \frac{\text{illegale Trefferquote}}{\text{illegale Trefferquote} + 1000 \cdot \text{legale Fehlerquote}} \tag{4.23}$$

4.5.3 Unparitätisches Training

Die niedrigen Konfidenzwerte erfordern eine weitere Optimierung der bei der Klassifizierung durch die RBF-Netze erzielten Trefferquoten.
Eine gleichmäßige Auswahl der Trainingsdaten im Verhältnis 1:1 aus den vorliegenden Klassen spiegelt nicht die in der Realität vorliegende Dominanz der

legalen Daten wieder. Es müßte von einem realen Verhältnis aus 1000:1 legalen und illegalen Transaktionen ausgegangen werden. Ein Training bezüglich dieses Verhältnisses aus legalen und illegalen Daten scheitert jedoch zum einen an der eingeschränkten Datenmenge, will man wiederholtes Training auf bereits benutzten Datensätzen vermeiden, und zum anderen an dem hohen Laufzeitaufwand, der für solch ein Training notwendig wäre. Es müssen ausreichend illegale Datensätze dem Netz präsentiert werden, um eine ausreichende Mißbrauchserkennung zu erreichen. Dies wiederum hat zur Folge, daß die Anzahl der legalen Datensätze auf ein entsprechend hohes Maß ansteigen würde. Aus diesem Grund werden kleinere, moderate Trainingsverhältnisse aus legalen und illegalen Daten ausgewählt, so daß mit einem minimalen Umfang an Trainingsdaten eine ausreichend gute Mißbrauchsanalyse möglich ist und trotzdem gute Trefferquoten seitens der legalen Datensätze gegeben sind.
In Tabelle 4.7 sind die Ergebnisse der Klassifizierung für verschiedene Trainingsverhältnisse auf Basis von 300 Trainingszyklen zusammenfassend dargestellt. Die Anzahl von 300 Trainingszyklen wurde aufgrund der guten Klassifizierungsergebnisse beim paritätischen Training gewählt. Die Verhältnisangaben in Spalte 1 bedeuten am Beispiel 2:1, daß für jeden illegalen Datensatz zwei legale zum Training verwendet werden. Das heißt, es werden insgesamt 100 illegale und 200 legale Trainingsdatensätze zum Lernen verwendet. Der anschließende Verifikationslauf auf insgesamt 500 Datensätzen erfolgt wieder in einem ausgewogenen Verhältnis von 1:1. Es werden also gleichermaßen legale und illegale Datensätze untersucht.

Verhältnis	Trefferquote	richtige Zuweisungen		falsche Zuweisungen		Konfidenz
		legale	illegale	legale	illegale	
2:1	78,8	95,2	62,4	4,8	37,6	1,283
3:1	78,2	98,4	58,4	1,6	41,6	3,522
4:1	58,2	99,6	16,8	0,4	83,2	4,031
5:1	52,5	99,2	6,0	0,8	94,0	0,744
10:1	50,0	100,0	0,0	0,0	100,0	100,0

Tabelle 4.7: Klassifizierungsergebnisse nach unparitätischem Training (in %)

Wie in Tabelle 4.7 zu erkennen ist, kommt es mit zunehmender Anzahl an legalen Trainingsdatensätzen schnell zu einer Steigerung der Trefferquote bei den legalen Einstufungen in der Verifikationsphase. Daraus resultiert zu einem großen Teil auch die Konfidenzsteigerung, die ebenfalls in Tabelle 4.7 abzulesen ist. Die aus der Reihe schlagende Konfidenz bei dem Verhältnis 5:1 ist auf eine eventuell ungünstige Datengrundlage bei der Verifikation zurückzuführen. Jedoch ist gleichzeitig ein Verlust an illegaler Trefferquote zu verzeichnen. In Zeichnung 6.7 auf Seite 145 ist diese Beziehung aus legalem und illegalem Zuweisungserfolg graphisch dargestellt. In diesem Zusammenhang verschiebt sich die eingezeichnete Klassengrenze hin zu `B`, was eine Konfidenzsteigerung zur Folge hat, wie man

ebenfalls an der Konfidenzkurve in Abbildung 6.7 erkennen kann.

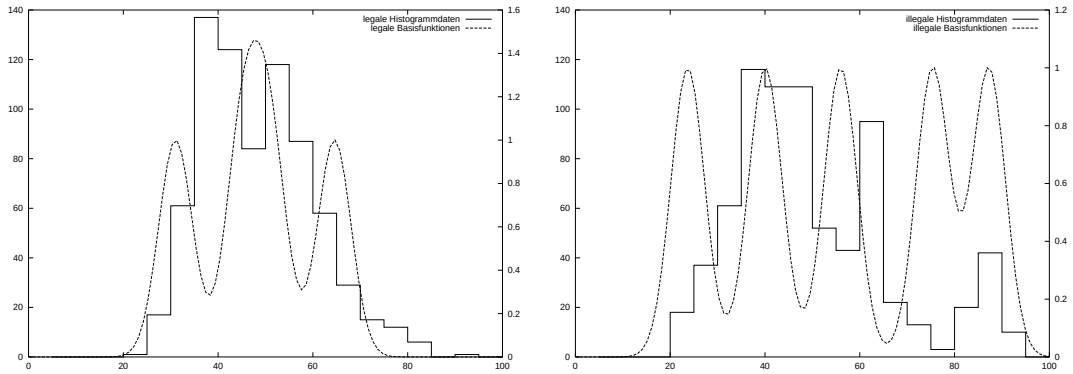
4.6 Rückverfolgung der Entscheidungsfindung

Wie erwähnt, ist es mit Hilfe der RBF-Netzwerke möglich, die Klassenentscheidungen auf Basis der einzelnen radialen Basisfunktionen zurückzuverfolgen und eine grobe Auswertung der dem Training und damit allen zugrundeliegenden Daten zu erhalten. Für die Auswertung wird dafür die Summe der einzelnen Basisfunktionen gebildet. Anhand der verschiedenen Extremwerte dieser Summenfunktion können dann Rückschlüsse auf die von den einzelnen RBF-Neuronen repräsentierten Datenbereiche gezogen werden. Je nach Klassenzugehörigkeit der Neurone kann eine grobe Unterteilung des Eingaberaumes bezüglich der Klassen durchgeführt werden.
Anhand einiger Beispiele soll diese Rückverfolgung nun nachvollzogen und beurteilt werden. Es wurden für diese Art der Auswertung nur 1-dimensionale Daten betrachtet, da die Darstellung der Basisfunktionen im mehrdimensionalen Raum nur schwer realisierbar und unüberschaubar ist. Als Beispiel dienen demzufolge die Transaktionsuhrzeit, das Karteninhaberalter und das Verhältnis aus Kreditrahmen und Transaktionsbetrag. Zum Vergleich mit den Orginaldaten werden diese in Form von Histogrammen getrennt nach Klassenherkunft dargestellt. Es werden also die Basisfunktionen, die die legalen Transaktionen repräsentieren mit den Histogrammen der legalen Daten dargestellt. Für die illegalen Basisfunktionen gilt dies entsprechend. Im Fall der Auswertung des Kreditrahmens sind jeweils beide Histogramme und Basisfunktionskurven dargestellt, so daß ein direkter Vergleich möglich ist.

4.6.1 Karteninhaberalter

Die Histogramme der Altersangaben der Karteninhaber ähneln sich in großem Maße, wie in Abbildung 4.4 zu erkennen ist. Eine unterschiedliche Ausprägung der Histogrammdaten ist nur im oberen Altersbereich um 90 Jahre festzustellen, wo ein auffällig hoher Anteil an Mißbrauchsdaten registriert werden kann. Trotzdem ist eine Klassifizierung nur sehr schwer durchzuführen. Dies ist auch an den Trefferquoten in Tabelle 4.4 in Spalte E zu erkennen. Die Trefferquote übersteigt nur knapp den Wert von 50%.
Anhand der legalen Daten erkennt man deutlich die Übereinstimmung der Summenfunktion der Basisfunktionen und dem Histogrammverlauf. Auch für den illegalen Datenteil ist festzustellen, daß der komplette Datenraum, der mit illegalen Transaktionsdatenwerten abgedeckt ist, auch von den entsprechenden Basisfunktionen erreicht wird. Es ist festzustellen, daß sich zum Großteil die Basisfunktionen über den Eingaberaum hinweg abwechseln. Dadurch kommt es zu widersprüchlichen Gegensätzen in Form der Extrempunkte von Summenfunktion und

Histogramm. Dies ist dadurch zu erklären, daß die ursprünglich angelegten Basisfunktionen, die die Aufgabe haben, die einzelnen Datenräume so gut wie möglich abzudecken, im Laufe des Trainings von der eigentlichen, initialen Zentrumsposition der RBF-Neuronen anderer Klassenzuordnungen verdrängt wurden. Dies ist unter den gegebenen Daten möglich, da sich legale und illegale Daten nur sehr wenig in der Ausprägung unterscheiden. Es kommt also im Laufe des Trainings zu einer bestmöglichen Verteilung der einzelnen Neuronen über dem gesamten Eingabedatenraum, um den Fehler bei der Klassifizierung so klein wie möglich ausfallen zu lassen.
Im Falle des Altersbereichs um die 90 Jahre ist eine solche Konkurrenz bei der Zuordnung durch den prägnanten Unterschied aus legalen und illegalen Daten nicht gegeben. Die Basisfunktionen, die diesen Bereich abdecken, liegen sehr nah zusammen, da kein legales Neuron zusätzlich in diesem Bereich angelegt wurde, daß zu einer Verdrängung geführt hätte. Auch im Altersbereich von 30 bis 55 Jahren, in dem ein vermehrter Anteil an legalen Transaktionen zu verzeichnen ist, der jedoch den ebenfalls hohen Anteil an Mißbrauchstransaktionen in diesem Bereich übersteigt, ist eine Dominanz der legalen Basisfunktionen zu verzeichnen. Auch hier kann eine eindeutige Einstufung durch die zugrundeliegenden Basisfunktionen gewährleistet werden. Die zentrale, legal markierte Basisfunktion im Bereich um das Alter von 50 Jahren ist gesäumt von illegal markierten Basisfunktionen, die von der zentralen Basisfunktion aufgrund der überwiegend legalen Transaktionsdaten im Laufe des Trainings an die Seiten gedrängt wurden. Es schließen sich unmittelbar wieder legale Basisfunktionen rechts und links an.



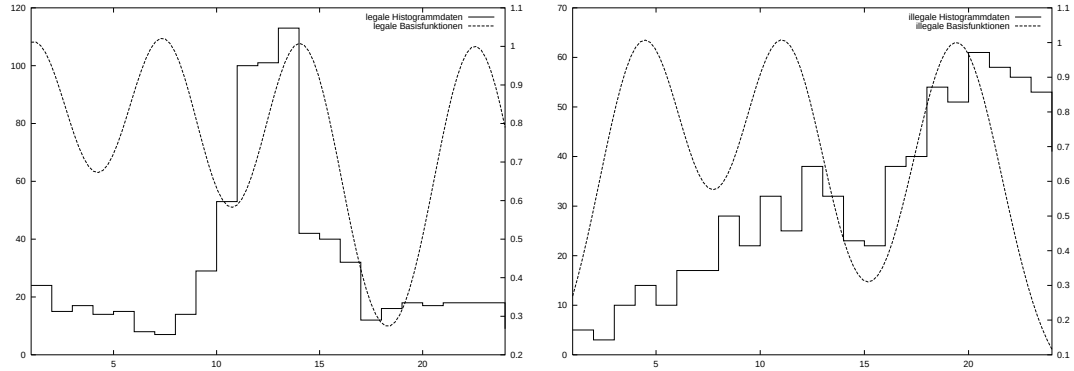
legale Datengrundlage illegale Datengrundlage

Abbildung 4.4: Gegenüberstellung von Datenverteilung und Basisfunktionen im Falle der Altersangaben

4.6.2 Transaktionsuhrzeit

In Abbildung 4.5 ist eine Übereinstimmung der Histogramme im Vergleich mit den Summenfunktionen der Basisfunktionen etwas besser zu erkennen. In dem

Zeitraum von 0 Uhr bis 5 Uhr scheint es, werden nur wenige Mißbrauchstransaktionen getätigt. Die Summenfunktion der Basisfunktionen im Falle der legalen Neuronen besitzt einen Hochpunkt. Ebenfalls ist der Zeitraum von 8 Uhr bis 15 Uhr im Histogramm eindeutig durch vermehrt legale Transaktionen charakterisiert. Auch die Summenfunktion der legalen Basisfunktionen erreicht hier erneut ein Maximum. Dieses Maximum scheint im Laufe des Trainings von einem illegalen RBF-Neuron getrennt worden zu sein, das für eine erhöhte Abdeckung der Mißbrauchsdaten in Folge einer ebenfalls festzustellenden Erhöhung der Mißbrauchstransaktionenanzahl in diesem Bereich sorgt. Dieses illegal zugeordnete Neuron verdrängt damit die legalen Basisfunktionen auf die umliegende Nachbarschaft, was zum Beispiel an der leicht nach rechts versetzten Basifunktion bei 14 Uhr zuerkennen ist. Je weiter die Transaktionsuhrzeit fortschreitet, umso stärker nimmt der Mißbrauch zu, was auch an dem dritten Maximum um den Wert von 20 Uhr der als illegal markierten Summenfunktion zu erkennen ist. Dennoch ist diese Basisfunktion ausgeprägt von den übrigen die illegalen Datenvertretenen Basisfunktion getrennt, wodurch auch die zugrundeliegenden Histogrammdaten repräsentiert werden. Ein ähnliches Verhalten ist bei den legal markierten Basisfunktionen zu erkennen.



legale Datengrundlage illegale Datengrundlage

Abbildung 4.5: Gegenüberstellung von Datenverteilung und Basisfunktionen im Falle der Uhrzeitdaten

Trotz der ersichtlichen Parallelen ist auch hier zu erkennen, daß sobald sich die legalen und illegalen Datenverteilungen aneinander angleichen oder sich überschneiden, es zu *keiner* zuverlässigen Klassifizierung mehr kommen kann. Eine genaue Positionierung und Ausdehnung der RBF-Neurone ist nicht mehr optimal für die zugrundeliegenden Daten möglich. Auch eine Spezialisierung durch eine Varianzverminderung kann hier nicht wesentlich zu einer Verbesserung führen. In diesem Fall würden sich die verschieden ausgezeichneten Basisfunktionen stärker abwechselnd über den Eingaberaum verteilen. Einzelnde Datenbereiche könnten zwar besser hervorgehoben werden, doch leidet dadurch die Abstraktionsgüte des RBF-Netzes.

4.6.3 Transaktionsbetrag in Relation zum Kreditrahmen

Die Ergebnisse dieser Auswertung sind in Abbildung 4.6 abgebildet. Durch die hohe Streuung der Transaktionsdaten kommt es zum Teil zu einer einigermaßen genauen Übereinstimmung der Histogrammdaten und den Summenfunktionen der Basisfunktionen. Am deutlichsten wird diese Übereinstimmung im oberen Bereich, wo strikt voneinander getrennt ausschließlich, wenn auch nur wenige, legale und illegale Daten auftreten. Diese Bereiche um die Werte 2100 im legalen und 1700 im illegalen Fall werden ebenso eindeutig von den entsprechenden RBF-Neuronen, beziehungsweise den Basisfunktionen abgedeckt.

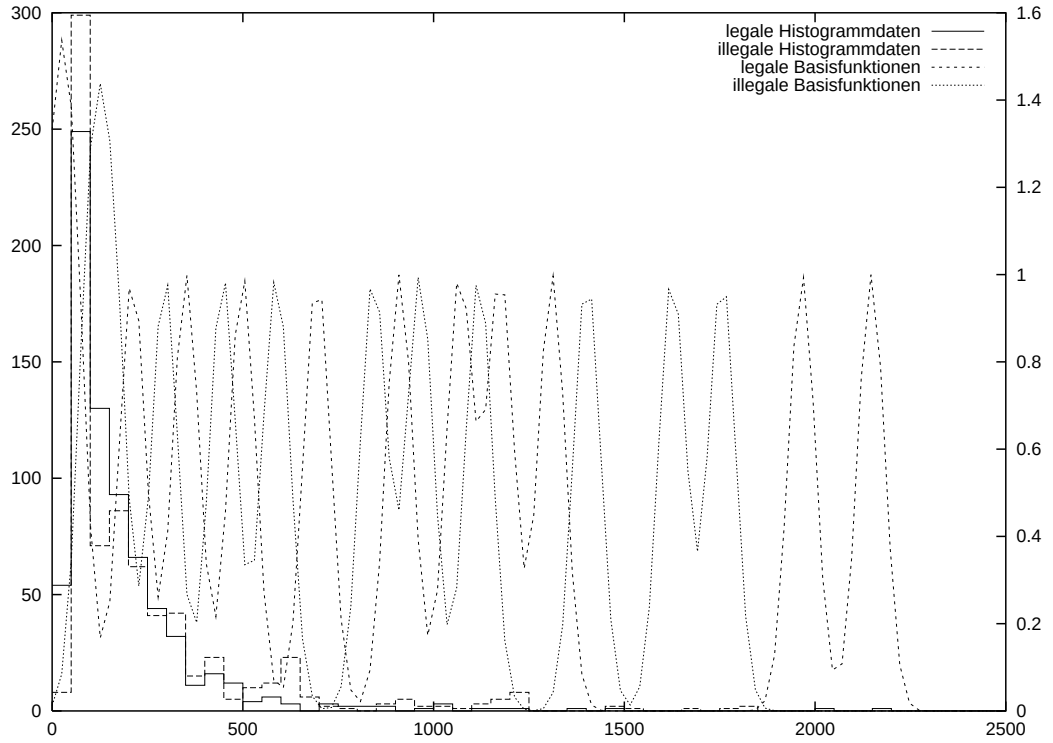


Abbildung 4.6: Gegenüberstellung von Datenverteilung und Basisfunktionen im Falle der Auswertung von Kreditrahmen und Transaktionsbetrag

Erst im unteren Teil, indem ein Gros der Daten vertreten ist, kommt es zu Überschneidungen und Überdeckungen der einzelnen radialen Basisfunktionen. Dennoch können auch hier Parallelen zwischen den Summenfunktionen und den Histogrammverläufen entdeckt werden. Zum Teil werden Maxima der Summenfunktionen von den Maxima der anderen Klasse getrennt. Trotzdem ist besonders bei den kleinen Werten gut die abwechselnde Vertretung der RBF-Neurone zu erkennen, die der ebenfalls wechselnden Mächtigkeit an legalen und illegalen Transaktionen entspricht. Es gilt auch hier, daß die zum großen Teil vorliegende Übereinstimmung der Mächtigkeiten der Transaktionsklassen die Klassifizierung

erschwert. Dies hat ebenfalls zur Folge, daß die Basisfunktionen nur in wenigen Fällen eindeutig Rückschlüsse auf die zugrundeliegenden Daten zulassen.

4.6.4 Zusammenfassung

Wie die Auswertung zeigt, ist es schwer eine Klassifizierung auf Basis der vorliegenden Daten durchzuführen, da ein Großteil der zugrundeliegenden Daten starke Übereinstimmungen zeigt. Dies kommt auch bei der rückführenden Auswertung der Basisfunktionen zutage. Hier können nur vage Rückschlüsse bezüglich der Datenverteilung der zugrunde liegenden Daten getroffen werden. Dennoch sind unter Kenntnis der Datenverteilungen, wie anhand der Histogramme deutlich wird, eindeutige Parallelen zwischen den Basisfunktionen und den Datenverteilungen festzustellen. Besonders vor dem Hintergrund dieser Verarbeitung und der Trainingsverfahren für die Basisfunktionen, ist es möglich, den Verlauf der Summenfunktion bezüglich einer Datenverteilung auszuwerten und Rückschlüsse ziehen zu können.

4.7 Ausblick und Verbesserungen

Da die Analyse eines neuen Verfahrens zur Mißbrauchsprävention durch die Kombination aus Data-Mining-Verfahren *und* neuronalen Netzen im Vordergrund dieser Diplomarbeit steht, wurde auf eine Optimierung der RBF-Netzarchitektur durch in der Literatur bekannte und vorgestellte Verfahren verzichtet. Dennoch soll an dieser Stelle auf einzelne Punkte eingegangen werden, die es unter Umständen erlauben, bessere Ergebnisse bei der Klassifizierung durch RBF-Netze zu erzielen.

So ist es zum Beispiel denkbar, im Laufe des Trainings wenig zur Klassifikation eingesetzte, das heißt selten aktivierte Neuronen aus der verdeckten Schicht zu entfernen. Diese Neurone vertreten damit nicht ausreichend repräsentative Daten und tragen damit nicht zur Abstraktion des Eingaberaumes bei. Durch die Entfernung dieser Neurone kann Rechenzeit zum Beispiel bei der Aktualisierung der Gewichte gespart werden, sowie Speicherplatz, der für die notwendigen, gespeicherten Parameter und Variablen benötigt wird. Eine andere Alternative in diese Richtung ist die Vereinigung benachbarter RBF-Neuronen der gleichen Klassenzuordnung. Auf diese Weise kann ebenfalls nach einer bestimmten Anzahl von Trainingsiterationen die Menge der Neuronen reduziert sowie das Abstraktionsniveau des Netzes erhöht werden.
Entsprechend [Pie95] können in diesem Zusammenhang auch generalisierte, radiale Basisfunktionen verwendet werden (GRBF), die sich dem Eingaberaum besser in ihrer Ausrichtung anpassen. Es handelt sich dabei statt um radialsymetrische Glockenfunktionen um ellipsoidale Funktionen. Dies ist vor allem ein Vorteil bei

mehrdimensionalen Eingaberäumen. Weitere Optimierungsverfahren sind diesbezüglich ebenfalls in [Pie95] ausführlich aufgeführt.
Eine weitere Möglichkeit, die neuronalen Netze zu verbessern, kann auf dem Wege einer weiter ausgebauten, spezialisierten Vorverarbeitung geschehen. Es ist somit denkbar, mit Hilfe anderer mustererkennender Systeme eine Vorklassifizierung der Eingabedaten oder Reduzierung der Eingabedimension vorzunehmen.

4.8 Zusammenfassung und Auswertung

Wie sämtliche Ergebnisse der Auswertung der Analogdaten auf Basis der RBF-Netze zeigen, kann alleine durch diese Art der Analyse nicht in ausreichendem Maße eine Mißbrauchsprävention stattfinden. Dies liegt zum einen an der sehr niedrigen Mißbrauchsquote von 0,1%, als auch an den zum Teil sehr ähnlichen Werteverteilungen der zugrundeliegenden Daten.
Erster Punkt führt dazu, daß selbst eine ausschließliche Klassifizierung der Transaktionen zugunsten der legalen Zuordnung zu einer 99,9% Erfolgsquote führen würde. Derartige Trefferquoten konnten bei weitem nicht mit den implementierten Netzwerken erreicht werden und ist außerdem im Falle von sich überlagernden Eingabemustern – wie vorliegend – auch nicht möglich. Damit bleibt als einzige Konsequenz, eine derartige Klassifizierung nur in Zusammenhang mit einem weiteren, den Datenraum einschränkenden oder vorklassifizierenden Verfahren zu bearbeiten, und somit eine bessere Grundlage für diese Art der Klassifizierung zu schaffen. Diese Kombination aus verschiedenen, eventuell gestaffelten Analyseverfahren wird eingehend in Kapitel 6 vorgestellt und behandelt.
Nicht desto trotz können mit dem implementierten Verfahren gute Trefferquoten um die 80% erreicht werden, wie in Tabelle 4.6 zu entnehmen ist, obwohl eine hohe Übereinstimmung der legalen und illegalen Werte, wie auch in Abschnitt 4.6 erwähnt, zu beobachten ist.
Durch das vorgestellte Verfahren in Form verschiedener konkurrierender RBF-Unternetze ist ein erweiterbares, anpassungsfähiges Modell zur Analyse der Analogdaten in den Transaktionen gegeben, das auf die jeweiligen Bedürfnisse zugeschnitten bei der Mißbrauchserkennung eingesetzt werden kann.

Kapitel 5

Profilauswertung

5.1 Motivation

Aufgrund der vielfältigen Einsatzweisen und Handhabungen von Kreditkarten, darf eine Profilanalyse der einzelnen Karteninhaber nicht fehlen. Eine benutzerspezifische Analyse der Gewohnheiten kann bei der Auswertung der Transaktionsdaten helfen, eine Entscheidung bezüglich des Karteninhaberverhaltens zu fällen. Es gibt verschiedene Arten, eine solche Profilanalyse zu modellieren.

5.1.1 Die Problematik

Bis zu diesem Zeitpunkt wurden die Daten benutzerunabhängig und zeitlich unberücksichtigt betrachtet und ausgewertet, das heißt es wurden weder spezielle Verhaltensmuster oder Gewohnheiten einzelnen Konteninhabern zugeordnet, noch die Daten in ihrer zeitlichen Aufeinanderfolge berücksichtigt oder über einen bestimmten Zeitraum verfolgt. Eine Einordnung von Transaktionen als Mißbrauchstransaktion kann durch eine Abweichung von den bisherigen Gewohnheiten des Kontoinhabers ausgelöst werden. Diese Vorgehensweise setzt jedoch ein benutzerspezifisches Profil für jeden Benutzer voraus. Dieser Ansatz wird vielfach eingesetzt, erfordert jedoch einen hohen verwalterischen Aufwand, der im Falle einer Implementierung den Umfang dieser Diplomarbeit übersteigen würde. Außerdem werden für diese Art des Profils ausreichend Transaktionsdaten von jedem Konto benötigt, um einigermaßen zuverlässig die Gewohnheiten des Karteninhabers und die kontenspezifischen Parameter bestimmen zu können.
Darum soll in diesem Zusammenhang eine Analyse der unmittelbar letzten Transaktionen verschiedener Konten realisiert werden, so daß es möglich sein wird, Verhaltensmuster zu vergleichen und diesbezüglich die Mißbrauchsentscheidung zu unterstützen. Dabei soll jedoch weiterhin generell kontounabhängig gearbeitet werden.
Weiter sind verschiedene Szenarien bekannt, die typisch für einen Mißbrauch sind,

wie zum Beispiel das „Abräumverhalten“, also der Versuch in möglichst kurzer Zeit möglichst viel von dem Konto abzuheben – darum im Fachjargon auch „Paniksyndrom“. Dies setzt ebenfalls eine zeitliche Analyse der Daten voraus, die bisher *so* nicht realisiert ist.

5.1.2 Profilanalyse auf den vorliegenden Daten

Wie bisher muß zwischen den verschiedenen vorliegenden Datentypen unterschieden werden. Spricht man bei einer Folge von symbolischen Werten von einer Zeitsequenz, so handelt es sich bei den analogen Daten um Zeitreihen. Eine differenzierte Betrachtungsweise ist also auch in diesem Zusammenhang notwendig.

Ein Benutzerprofil ist die von Redundanz befreite, individuelle Beschreibung der Transaktionshistorie bezüglich der einzelnen Konten. Um ein solches Profil erzeugen zu können, ist es notwendig, ausreichend Transaktionsdaten zur Analyse und Erstellung eines solchen Profils zur Verfügung zu haben. Aus diesem Grund wurde die Verteilung der Transaktionen auf die einzelnen Konten ermittelt, und somit die maximale Anzahl an zur Profilanalyse zur Verfügung stehenden Transaktionen bestimmt. Abbildung 5.1 zeigt die Verteilung der legalen Transaktionen der aktiven Konten. Dabei ist deutlich zu erkennen, daß für einen Großteil der

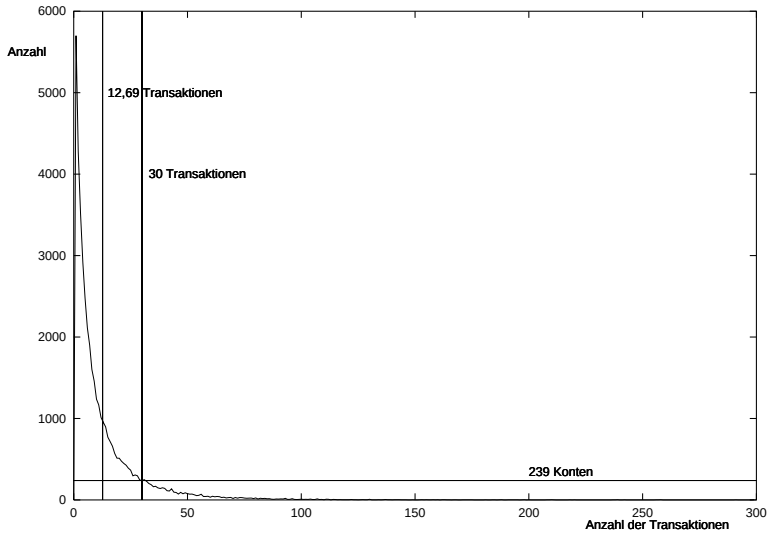


Abbildung 5.1: Legale Transaktionen pro Konto

legalen Konten weniger als 50 Transaktionen in den vorliegenden Daten zu verwerten sind. Eine Transaktionsanzahl von mehr als 300 Transaktionen konnte nur 2 mal erreicht werden. Die Untersuchung der aktiven Konten ergab im Mittel eine Transaktionsanzahl von 12,69 Transaktionen. Entsprechend fällt der Verlauf für die illegalen Transaktionen beziehungsweise die gesperrten Konten aus.

Demzufolge ist nur eine begrenzte Anzahl an Konten mit ausreichendem Material zur Erstellung eines passenden Benutzerprofils vorhanden. Dies gilt besonders vor dem Hintergrund, daß für die Verifikation sowohl ausreichend legale als auch illegale Transaktionen eines Kontos vorliegen müssen, was durch die Datenanordnung und den zu kurzen Zeitraum, der von den Beispieldaten abgedeckt wird, nicht gewährleistet ist. So ist für die Erstellung eines solchen Profils der Vorlauf zur Rekrutierung legaler Transaktionen zur Erstellung einer kontenspezifischen Transaktionsstatistik zu kurz, besonders, wenn zur Verifikation zusätzlich noch Mißbrauchstransaktionen zu dem entsprechenden Konto vorhanden sein sollen. Zu Simulationszwecken muß also in diesem Zusammenhang nach einer allgemeinen, kontounabhängigen Lösung gesucht werden.

5.2 Profilalternativen

Ein Benutzerprofil stellt im eigentlichen Sinne eine komprimierte, von überflüssigen Informationen befreite Charakterisierung der benutzerspezifischen Eigenschaften und Verhaltensweisen dar.
Neben Schwellen- und Mittelwertsprofilen bietet sich im vorliegenden Fall auch an, die Transaktionshistorie mit Augenmerk auf das oben beschriebene Abräumverhalten zu beobachten. Diese Art von Analyse erinnert an eine Zeitreihenanalyse. Es wird in Abhängigkeit von der zeitlichen Reihen- beziehungsweise Aufeinanderfolge der zu untersuchenden Datenwerte eine Entscheidung bezüglich des Mißbrauches getroffen.
Im folgenden soll nun untersucht werden, auf welche Weise eine solche Verarbeitung für das vorliegende Problem der Klassifizierung auf den vorliegenden Daten realisiert werden kann. In Abschnitt 5.2.1 wird diesbezüglich zunächst auf die symbolischen Daten eingegangen. Es werden einige Modelle in diesem Zusammenhang erläutert, die jedoch teilweise auf Grund eines zu hohen Aufwands verworfen werden müssen. Anschließend folgt in Abschnitt 5.3 die Erörterung eines Modells zur Auswertung von Zeitreihen auf den Konten in Anlehnung an Kapitel 4. Die Ergebnisse, die mit den verwendeten Verfahren erzielt werden können, werden anschließend in Abschnitt 5.5 vorgestellt und in Abschnitt 5.5.2 ausgewertet und diskutiert.

5.2.1 Profile auf Basis der symbolischen Daten

Um Zeitsequenzen – bestehend aus symbolischen Daten – auszuwerten, sind verschiedene Vorgehensweisen denkbar. Ein Beispiel für eine solche Zeitsequenz von verschiedenen symbolischen Daten könnte zum Beispiel, wie in Abbildung 5.2 gezeigt, aussehen. Hier sind die Zeitsequenzen zweier Datentypen (i_1 und i_3) über den Zeitraum t_1 bis t_5 dargestellt. Δ_i bezeichnet dabei die Zeitdifferenz zwischen den einzelnen Transaktionen.

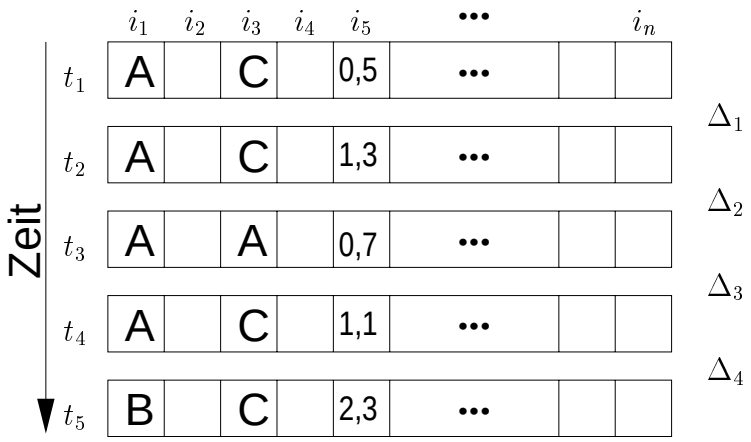


Abbildung 5.2: Beispiel von Zeitsequenzen

Im folgenden sollen nun verschiedene Modelle in Hinblick auf solche Zeitsequenzen diskutiert und auf ihre Verwertbarkeit für die vorliegende Problematik hin untersucht werden.

Das Markov-Ketten-Modell

Eine Alternative zur Auswertung der symbolischen Daten beruht auf der wahrscheinlichkeitsgestützten Auswertung der einzelnen Sequenzen, indem die Übergangswahrscheinlichkeiten[1] von einem Zustand in den folgenden bestimmt werden. Dies kann mit Hilfe von Markov-Ketten realisiert werden [Pfl86]. Dabei wird die Wahrscheinlichkeit berechnet, einen bestimmten Zustand zu erreichen in Abhängigkeit der m-letzten Zustände; man spricht von der „m-stufigen Markov-Eigenschaft“. Es handelt sich dabei um eine Kette von Übergangswahrscheinlichkeiten, die sich zu einer Mißbrauchstransaktion hin entwickeln, so daß frühzeitig auf einen Mißbrauch geschlossen werden kann.
Formal stellt eine Markov-Kette eine auf einem endlichen Zustandsraum $U = \{u_1, \ldots, u_j\}$ basierende Folge von Zufallswerten aus X mit den Zufallsvariablen $x_1, \ldots, x_n$ dar, und es gilt:

$$P(X_n = u_j | X_1, \ldots, X_{n-1}) = P(X_n = u_j | X_{n-1}) \qquad \text{mit} \qquad m = 1 \tag{5.1}$$

Dabei kann auch hier zunächst von zeitunabhängigen Übergangswahrscheinlichkeiten ausgegangen werden, so daß *homogene Markovketten* verwendet werden können. Das heißt, die Wahrscheinlichkeit $p_{ij}^t := P(X_n = u_j | X_{n-1} = u_i)$ hängt *nicht* von n ab. Damit läßt sich aus den Übergangswahrscheinlichkeiten p_{ij}^t eine quadratische Übergangsmatrix $\mathbf{P^T}$ erstellen.

[1] *engl.*: transition probability

Da jedoch die Übergangswahrscheinlichkeiten für **jeden** Wert eines Zustands u_i zu *jedem* Wert eines potentiellen nächsten Zustandes u_{i+1} berechnet werden müssen, kommt es bei diesem Verfahren zu einem hohen Speicher- und Rechenaufwand, zumal einige symbolische Datenfelder mehrere 10.000 bis 100.000 Werte w annehmen können (siehe Tabelle B.2). Das bedeutet, daß alleine die Übergangsmatrix P^T insgesamt w^2 Übergangswahrscheinlichkeiten speichert. Je nach Anzahl der Stufen m, beziehungsweise nach Größe des zu betrachtenden Fensters gilt dann $m \cdot w^2$. Am Beispiel des Händlercodes (`MER_ID`)[2] folgt daraus, daß $3 * 138765^2 = 5,78^{10}$ Übergangswahrscheinlichkeiten p_{ij}^t bestimmt werden müssen. Außerdem darf die Tatsache nicht außeracht gelassen werden, daß eine solche Übergangsmatrix $\mathbf{P^T}$ dann für jeden Karteninhaber bestimmt werden muß, man spricht von der „kombinatorischen Explosion“. Eine derartige Matrix in Verbindung mit jedem Benutzer zu bestimmen, rechtfertigt nicht den Aufwand, den ein solches Verfahren in Bezug auf Speicheranforderung und Rechenkapazität erfordert. Dennoch wurde dieses Verfahren mit guten Ergebnissen bei der Spracherkennung eingesetzt, jedoch wurde auch hier schon auf die hohe Komplexität hingewiesen [Bra95].
Der Vorteil dieses Systems liegt darin, daß, sobald ein Trend der untersuchten symbolischen Daten mit ausreichender Wahrscheinlichkeit auf einen Mißbrauch hindeutet, eingeschritten und das Konto frühzeitig gesperrt werden kann.

Statistische Auswertung

Eine andere Alternative bietet die kontospezifische Berechnung der einzelnen Wahrscheinlichkeiten $P(X = x_j)$ **aller** Werte der profilrelevanten, symbolischen Datentypen. Diese werden dann zu einer Gesamtwahrscheinlichkeit berechnet, und mit den kontospezifischen, typischen Transaktionswerten verglichen. Dabei spielt die Art der Berechnung keine entscheidende Rolle, vielmehr ist ein einheitliches Maß von Bedeutung, das zum Vergleich herangezogen werden kann.
Weicht bei einer Vergleichsregel die Gesamtwahrscheinlichkeit zu weit von der benutzerspezifisch berechneten Wahrscheinlichkeit ab, so kann die Transaktion als untypisch für den Benutzer gewertet und somit als Mißbrauch gedeutet werden. Es wird also eine vorgegebene Schwelle vorausgesetzt, die jedoch für jeden Karteninhaber individuell gesetzt werden muß, da unterschiedliche Verhaltensmuster der einzelnen Karteninhaber obligatorisch für derartige Schwellenüberschreitungen wären.
Eine weitere Möglichkeit, diese Schwelle benutzerabhängig zu gestalten, kann durch die Berechnung der Standardabweichung σ und dem Mittelwert μ erreicht werden[FP97]. Dazu wird der berechnete Gesamtwahrscheinlichkeitswert $\sum P$

[2] siehe Anhang B

wie in Gleichung 5.2 eingesetzt:

$$\text{Ausgabe} = \begin{cases} \sum P & \text{wenn} \quad \sigma = 0 \\ \frac{\sum P - \mu}{\sigma} & \text{wenn} \quad \sum P > \mu \\ 0 & \text{sonst} \end{cases} \tag{5.2}$$

Der Nachteil bei dieser Vorgehensweise ist, daß zwar wesentlich weniger Wahrscheinlichkeiten als im vorhergehenden Fall, aber immer noch sehr viele pro Konto gespeichert werden müssen. Weiter kann mit diesem Verfahren kein Mißbrauch erkannt werden, der aus „Normalwerten“, also den üblichen Zustandswerten besteht. Hinzu kommt, daß eine zeitliche Analyse nicht vorgesehen und ohne weiteres möglich ist, so daß das Abräumverhalten nicht aufgedeckt werden kann. Die Bestimmung der Wahrscheinlichkeiten erfolgt zeitunabhängig.
Als Beispiel jedoch für eine erfolgreiche Erkennung in diesem Zusammenhang diene hier der Kontoinhaber, der regelmäßig nur am Geldautomat abhebt und plötzlich ausschließlich „Online-“ Buchungen beim Kreditinstitut eintreffen; dies läßt auf einen möglichen Mißbrauch schließen.

Wiederholungen

Ein nicht unwesentlicher Faktor bei der Mißbrauchserkennung beruht auf der Tatsache, daß Mißbräuche oft durch gleiche, symbolische Daten in aufeinanderfolgenden Transaktionen auffallen. Es wurde versucht, diese Alternative, die auf Erfahrungen der Fachkräfte der GZS basiert, algorithmisch zu realisieren und zusätzlich zur analogen Profilanalyse zu implementieren. Es wird davon ausgegangen, daß eine auffällige Anhäufung von gleichen Transaktionsdaten ein mißbrauchsspezifisches Kriterium ist.
So kann zum Beispiel ein mehrmaliges Buchen in Folge über das Internet allein per Kartennummer[3] auf einen Mißbrauch mit der Karte hinweisen. Die Beobachtung dieses Zusammenhangs wird erfolgreich bei der Expertenanalyse in Zuge der derzeitigen Mißbrauchsanalyse eingesetzt. Natürlich ist es eine Ermessensfrage, wo die Grenze zwischen den zeitlich aufeinanderfolgenden Transaktionen mit gleichen Werten gezogen werden muß.
Weiter ist dabei jedoch auch folgendes Szenario zu beachten, daß gewisse Karteninhabergruppen ihre Karte nur für ein und den selben Zweck einsetzen, zum Beispiel für das Bezahlen an der Tankstelle, so daß zwangsläufig Übereinstimmungen in den Händlerdaten vorhanden sind. Jedoch kann die Zeitdifferenz in diesem Zusammenhang ebenfalls als entscheidendes Kriterium angesehen werden. Übersteigt diese Zeitdifferenz einen vorgegebenen Wert, so kann von einem normalen Gebrauch der Karte ausgegangen werden, es liegt kein potentielles Indiz

[3]Dies ist überhaupt ein sehr unsicheres Verfahren, da es den persönlichen Kontakt meidet, autorisierte Kartenlesegeräte nicht zum Einsatz kommen und eine Unterschrift **nicht** getätigt werden muß und letztendlich nicht einmal die Geheimnummer verlangt werden kann.

Merkmalsnr.	Bezeichnung	legal		illegal	
		#	P in (%)	#	P in (%)
1.	ACCT_NBR[5]	50	*100,00*	50	*100,00*
3.	TRN_TYP	24	35,04	49	71,54
6.	CURR_CD	29	45,82	50	79,00
7.	POS_ENT_CD	19	18,62	30	29,40
12.	CRD_TYP[5]	50	*100,00*	50	*100,00*
13.	ICA_CD	36	44,64	26	32,24
14.	AID_CD	21	19,74	26	24,44
15.	SIC_CD	8	5,28	25	16,5
16.	ACT_CD	48	90,24	46	86,48
17.	MSG_TYP	28	43,68	50	78,0
18.	MER_ID	1	0,34	16	5,44
19.	MER_CNTY_CD[5]	50	*100,00*	50	*100,00*

Tabelle 5.1: Wiederkehrendes Auftauchen von symbolischen Daten

für einen Mißbrauch vor.

Um diesem Mißbrauchskriterium auf den vorliegenden Daten zunächst Gültigkeit zu bescheinigen, wurde auf einer Menge von 50 legalen sowie illegalen Datensätzen eine Statistik der Wiederholungen von Werten auf den in Frage kommenden Merkmalen erstellt. Die Ergebnisse sind in Tabelle 5.1 aufgeführt. Dafür wurden die Wiederholungen für eine Fenstergröße von 3 Datensätzen bestimmt. Das heißt, es wurden jeweils 3 Transaktionen eines Kontos auf Gleichheit der symbolischen Werte untersucht.
Um ein Maß für die Einstufung der Ergebnisse zu haben, also eine Gewichtung der einzelnen Merkmale in diesem Zusammenhang, soll auch die abhängige Wahrscheinlichkeit P entsprechend Gleichung 5.3 berechnet werden, mit der eine Einstufung auf Basis der Wiederholungen in die jeweilige Klasse möglich ist. Die Ergebnisse dieser Wahrscheinlichkeitsberechnung sind ebenfalls in Tabelle 5.1 enthalten. Die Wahrscheinlichkeitswerte werden dabei folgendermaßen berechnet: Der Wahrscheinlichkeitsraum S mit $S = 100$ getesteten Transaktionstripeln[4], jeweils 50 aus den legalen und illegalen Transaktionen, dient als Grundlage der Berechnung. Weiter entspricht E dem Ereignis:„Es liegt eine (il)legale Transaktionssequenz vor“, und F bezeichnet in diesem Fall das Ereignis: „Es liegt ein Tripel, beziehungsweise eine n-stellige Folge von gleichen Werten vor“, wobei n

[4]Es wird eine Fenstergröße von 3 angenommen.
[5]Per Definition! Das Ergebnis beruht auf fixen Konteneigenschaften, die nur in Ausnahmefällen geändert werden.

der Fenstergröße entspricht. Weiter gilt:

$$P(E|F) = \frac{\#(\text{il})\text{legale Transaktionstripel}}{50}$$

und

$$P(F) = P(F|S) = \frac{\#\text{illegale Transaktionstripel} + \#\text{legale Transaktionstripel}}{100}$$

Damit ergibt sich dann die abhängige Wahrscheinlichkeit $P(E|S)$ mit

$$P(E|S) = P\left(\frac{E \cap F}{S}\right) = P(F|S) \cdot P(E|F) \tag{5.3}$$

Am Beispiel des Transaktionstyps (`TRN_TYP`) sei die Berechnung zur Demonstration durchgeführt:

$$\text{legale Wahrscheinlichkeit} = \frac{24}{50} \cdot \frac{73}{100} = 0,3504 = 35,04\%$$
$$\text{illegale Wahrscheinlichkeit} = \frac{49}{50} \cdot \frac{73}{100} = 0,7154 = 71,54\%$$

Die so berechnete Auftrittswahrscheinlichkeit gibt Auskunft über die Wahrscheinlichkeit, mit der die aus identischen Daten bestehenden Datentupel in den jeweiligen Datenklassen auftreten. Betrachtet man die Unterschiede der Auftrittswahrscheinlichkeiten, so erkennt man die für die Analyse geeigneten Datensätze. So kann zum Beispiel im Falle des Transaktionstyps häufig eine Folge von gleichen Werten festgestellt werden, in Hundert Fällen insgesamt 73 mal, jedoch übertrifft auch die Anzahl der Auftritte in den illegalen Datensätzen einer solchen Datensequenz die Anzahl der Auftritte in den legalen Daten. Ein solches Datum ist für die Mißbrauchsanalyse folglich geeignet. Anders verhält es sich zum Beispiel im Falle des Datums `AID_CD`, bei dem die Auftrittswahrscheinlichkeiten in beiden Klassen nahezu übereinstimmen. Dieses Datum spielt also *keine* Rolle im Zusammenhang mit der Mißbrauchsanalyse bei diesem Verfahren.

Kommentar zu den statistischen Verfahren

Von der allgemein üblichen Methode, jedem Benutzer ein auf Basis der symbolischen Daten beruhendes Profil zuzuordnen und bei einer Abweichung von diesem Profil auf Mißbrauch zu entscheiden, soll hier bewußt verzichtet werden, da der Aufwand, für jedes Konto ein solches Profil, wie oben beschrieben, anzulegen, als zu aufwendig und die Kosten[6] als zu hoch erachtet werden. Außerdem liegt der Schwerpunkt dieser Diplomarbeit auf der allgemeinen Mißbrauchserkennung, also

[6]Im Sinne von Speicherplatz, Berechnungs- und Zugriffszeit

auf Basis aller Daten, so daß eine derartig Realisierung den Umfang der Arbeit sprengen würde.
Will man statt dessen allgemein, also kontounabhängig, die Zeitsequenzen mit dem Modell der Markov-Ketten auswerten, so erfordert dies neben dem immensen Rechenaufwand für die Berechnung der Übergangswahrscheinlichkeiten p_{ij}^t auch eine Speicherung der zurückliegenden Transaktionen bei den zu überwachenden Zuständen pro Konto, um die Übergangswahrscheinlichkeiten zu berechnen. Aus diesem Grund soll auch dieser Ansatz nicht weiter verfolgt werden.
Geht man vom wahrscheinlichkeitsgestützten Modell der statistischen Auswertung auf Basis aller Konten aus, so läuft das Ergebnis auf die Auswertung durch die Generalisierung aus Abschnitt 3.9 hinaus. Hier werden die einzelnen Wahrscheinlichkeiten der verschiedenen symbolischen Werte auf der Gesamtmenge der Transaktionen bestimmt, gespeichert und zur Klassifizierung herangezogen.
Außer auf die Analyse von Wiederholungen bei den symbolischen Daten soll deshalb auf eine zeitabhängige, kontospezifische Analyse dieser Art von Daten verzichtet werden.

5.2.2 Profile auf Basis der analogen Daten

Es bleibt die zeitliche Analyse der analogen Daten. Diese analogen Daten bilden mit den entsprechenden Datumswerten eine sogenannte Zeitfunktion über einem Intervall, das durch das vorgegebene Zeitfenster festgelegt ist. Als Beispiel diene hier ebenfalls die Abbildung 5.2 auf Seite 96. Diesmal wird Bezug auf den Datentyp i_5 genommen, bei dem es sich um einen analogen Datentypus handelt.

Um den Aufwand eines strengen Benutzerprofils zu sparen, wurde nach Wegen gesucht, eine allgemeinere, automatisierte Vorgehensweise zu finden, die es erlaubt, verlaufsbedingte Mißbrauchsvorgänge kontounabhängig zu erkennen. Es soll also auch bei den Analogdaten wie bei den symbolischen Daten darauf verzichtet werden, Benutzerprofile in Form von Mittelwerten und verallgemeinerten Daten in Bezug zu den einzelnen Konten zu speichern.

Um das von Experten erklärte „Abräumverhalten“ algorithmisch aufzuspüren, beziehungsweise einem Netz anzulernen, wird nun versucht, diesem neuronalen Netz mit Hilfe eines Zeitfensters über den Daten, Buchungsabläufe anzutrainieren. Die Zeitfunktionen über dem Fenster einer zuvor bestimmten Transaktionenanzahl können nun einem neuronalen Netz entsprechend einem Sollwert antrainiert werden. Ist in dem Zeitfenster ein Mißbrauch aufgetaucht, so wird das gesamte Zeitfenster und damit der Funktionsausschnitt als charakteristisch für einen Mißbrauch angesehen. Mit dieser Vorgehensweise kann ein Szenario, wie es durch das Abräumverhalten gegeben ist, erkannt werden. Dabei kann das Fenster über den Transaktionen so gewählt werden, daß mit jeder neuen Transaktion die älteste aus dem Fenster entfernt wird. Es wird also immer nur ein Ausschnitt der

Zeitfunktion analysiert (siehe dazu auch Abbildung 5.3).

Man kann dabei erneut zwischen einer allgemeinen und einer kontenspezifischen Alternative entscheiden. Bei der allgemeinen Vorgehensweise kann ein Netzwerk auf die Transaktionensreihen sämtlicher Konten trainiert werden. Bei der kontenspezifischen Vorgehensweise werden nur die Transaktionen des jeweiligen Kontos dem Netz antrainiert, das heißt zu jedem Konto müssen die entsprechenden Netzparameter gespeichert werden. Dabei handelt es sich bei weitem nicht um die Menge, wie sie bei den symbolischen Daten im Zuge der Profilauswertung veranschlagt wurde. Es handelt sich in diesem Falle im wesentlichen um nur wenige Netzparameter und -gewichte.

5.3 Umsetzung der analogen Profilanalyse

Es wurde für die Klassifizierung der Zeitfunktionen erneut ein radiales Basisfunktionsnetz, ähnlich denen aus Kapitel 4, benutzt. Dieses wurde um einen entsprechenden Eingabefilter zur Verarbeitung der Zeitreihendaten ergänzt.
Dieser Eingabefilter bestimmt die Zeitdifferenzen zwischen den einzelnen Transaktionen eines Kontos und dient zur Pufferung der relevanten Daten. Diese und die entsprechenden normierten Transaktionsbeträge werden dem Teilnetz in Form eines Eingabevektors zur Verarbeitung eingespeist. Der Eingabevektor hat die Dimension: $[2 \cdot \text{Fenstergröße} - 1]$. In Abbildung 5.3 ist dieser Vorgang zusammenhängend, graphisch dargestellt. Dabei sind die analogen Zeitwerte als a_i bezeichnet, die Zeitdifferenzen erneut mit Δ_i.

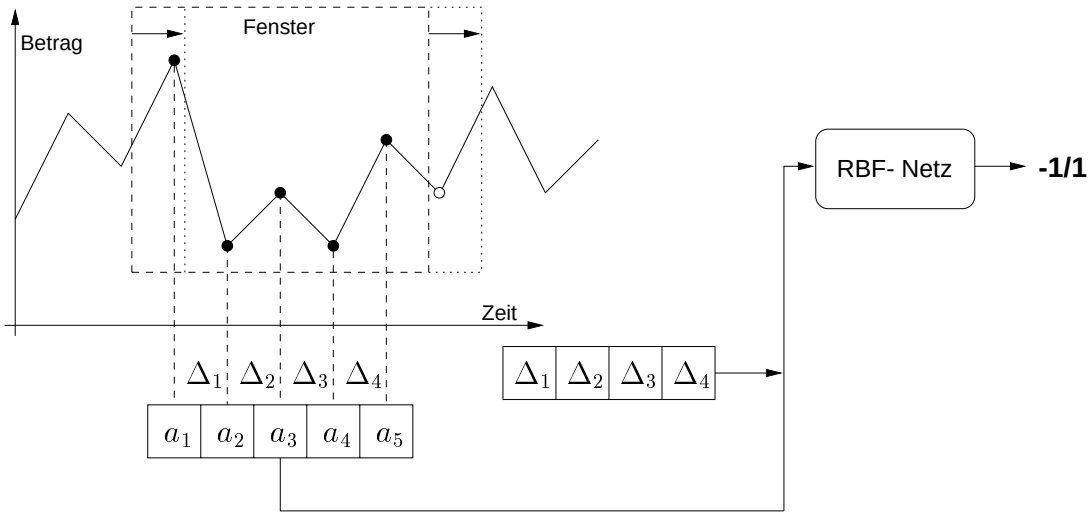


Abbildung 5.3: Zeitreihenanalyse

Dieser Vektor, bestehend aus Zeitdifferenzen und Betragswerten, wird nun als Eingabe der RBF-Netzinstanz zur Verarbeitung gegeben. Wie zuvor auch, kommt

es abschließend zu einer binären Klassenentscheidung des Netzwerkes.

5.3.1 Verrechnung der Betragswerte

Da es sich bei diesem Ansatz um die Realisierung einer allgemeinene Profilerstellung handelt, die also nicht kontenabhängig ist, muß aufgrund der verschiedenen Benutzerverhalten eine Normierung der verwendeten Daten stattfinden. Dies ist notwendig um den Eingabemusterraum ein wenig einzuschränken, da sonst eventuell zu spezielle Eingabewerte eine Abstraktion bei der Auswertung verhindern. Als Beispiel diene der Transaktionsbetrag verschiedener Karteninhaber, der zwar ebenfalls kontenspezifisch variiert, jedoch je nach Konto unterschiedliche Betragsregionen erreicht. Das Umsatzvolumen des Kontos fällt unterschiedlich hoch aus. Ähnlichkeiten in den Betragsschwankungen, die auf einen Mißbrauch, beziehungsweise das Abräumverhalten hinweisen, müßten so einzelnd für jede dieser Betragsregionen gelernt werden. Dies verhindert eine Normierung der analogen Betragswerte.
Dazu wird der Mittelwert μ aus den zur Profilanalyse herangezogenen i Analogwerten a_i berechnet. Anschließend werden diese a_i jeweils durch den so bestimmten Mittelwert μ dividiert. Dieser Vorgang, formal in Gleichung 5.4 beschrieben, entspricht einer Normierung der Analogwerte.

$$\tilde{a}_i = \frac{a_i}{\mu} \qquad \text{mit} \qquad \mu = \frac{\sum_i a_i}{i} \tag{5.4}$$

5.3.2 Berücksichtigung der Zeitdifferenzen

Auch für die Zeitdifferenzen ist eine Vorverarbeitung, jedoch keine Normierung, notwendig. Statt dessen soll ein Bezug zum Transaktionsbetrag hergestellt werden. Für diesen Fall wird erneut der Mittelwert μ der einzelnen Transaktionsbeträge a_i herangezogen und durch eine Division mit dem jeweiligen Zeitwert Δ_j verrechnet. Auf diese Weise wird eine Beziehung zwischen Transaktionsbeträgen und Zeitdifferenzen hergestellt.
Vorstellbar sind zum Beispiel geringe Zeitdifferenzen. Diese werden zusätzlich mit einem geringen Betragsmittelwert, wie er für ein „normales“ Verhalten gewöhnlich ist, verrechnet. Kommt es hingegen zu einem hohen Betragsmittelwert, so werden auf diese Weise die berechneten Zeitdifferenzwerte stark verringert, so daß ein Unterschied zum Normalfall eintritt. Sind die Zeitdifferenzwerte jedoch selbst sehr hoch, so wird der hohe Betragsmittelwert dadurch wieder relativiert. Demzufolge lassen sich die letztendlich zur Analyse benutzten Zeitwerte folgendermaßen charakterisieren: Liegen kleine Werte vor, so ist die Möglichkeit eines Abräumverhaltens wahrscheinlicher als bei großen Werten.
Tabelle 5.2 macht diese Art der Verrechnung anhand eines Beispiels deutlich:

Buchungsbeträge	Zeitdifferenz Δ	Mittelwert μ	Berechnung[7]	Ergebnis
100,- DM 200,- DM	3 h	150	$\frac{3 \cdot 1000}{150}$	20
10.000,- DM 20.000,- DM	36 h	15000	$\frac{36 \cdot 1000}{15000}$	2,4
10.000,- DM 20.000,- DM	3 h	15000	$\frac{3 \cdot 1000}{15000}$	0,2

Tabelle 5.2: Beispiel zur Verrechnung der Zeitwerte

Die auf die beschriebene Art und Weise verrechneten Transaktionswerte werden gemeinsam einer RBF-Netzinstanz, wie sie in Kapitel 4 beschrieben ist, zur Auswertung übermittelt. Die endgültige Klassifizierung wirkt sich damit auf die letzte der betrachteten Transaktionen aus, da diese in Abhängigkeit zu ihrer Transaktionshistorie zu beurteilen ist.

5.4 Umsetzung der Zeitsequenzen

In einem von der analogen Profilauswertung abgekoppelten Verfahren wird eine Auswahl an symbolischen Transaktionsdaten entsprechend Abschnitt 5.2.1 auf Wiederholungen hin überwacht. Kommt es zu dem Fall, daß innerhalb einer vorgegebenen Zeit mehrere Datenwerte in Folge den selben Werte annehmen, so entscheidet dieses Modul auf Mißbrauch. Die Schwelle, ab wann auf Mißbrauch entschieden wird, muß benutzerseitig vorgegeben werden. Mit Hilfe der im Trainingsverlauf bestimmten Auftrittswahrscheinlichkeiten gemäß Gleichung 5.3 kann dann der Einfluß der einzelnen Datentypen auf diese Entscheidung angegeben werden. Die Auftrittswahrscheinlichkeiten werden entsprechend der Gewichte der RBF-Netze für zukünftige Vergleiche abgespeichert.
Ist der Zeitraum zwischen den Transaktionen jedoch größer als der festgelegte Zeitraum, so wird die Überwachungseinheit initialisiert und kein Hinweis auf einen möglichen Mißbrauch gegeben.
Auch für die Arbeitsphase gilt: Erreicht die Summe der Auftrittswahrscheinlichkeiten, die im Laufe des Trainings bestimmt wurden, einen zuvor festgelegten Wert, so kann auf Mißbrauch entschieden werden und das Analysemodul gibt diese Entscheidung in binärer Form ebenfalls entsprechend der Vorzeichenfunktion[8] sgn(.) aus.

[7] Zur besseren Anschaulichkeit wurde die Zeitdifferenz in Stunden gewählt und deswegen mit einem konstanten Faktor multipiziert, um unübersichtliche Dezimalstellen zu vermeiden.

[8] siehe Abschnitt 4.3.2

5.5 Ergebnisse der Profilanalyse

5.5.1 Trainings- und Testläufe

Nachdem die Profilanalyse der Analogdaten beziehungsweise die Zeitsequenzanalyse für die letzten Transaktionen eines Kontos mit der entsprechenden Fenstergröße abgeschlossen ist, werden die binären Ergebnisse in einem abschließenden, linearen Neuron verrechnet. Dort werden sozusagen die Ergebnisse der einzelnen Verarbeitungstypen gewichtet und zusammengefaßt. Anhand des Gewichtsvektors kann dann die Bedeutung und Aussagekraft der einzelnen Analyseverfahren für die gemeinsame Auswertung interpretiert werden.

Zunächst wurde jedoch das Profilnetz, bestehend aus den Analog- und Symboldaten analysierenden Modulen, trainiert. Dafür wurden 400 Trainingszyklen verwendet. Das heißt, es wurden jeweils 200 gesperrte beziehungsweise aktive Konten ausgewählt und die entsprechenden Mißbrauchstransaktionen und legalen Transaktionen zum Lernen der Gewichte und Auftrittswahrscheinlichkeiten verwendet. Es wurde eine Fenstergröße von 3 gewählt, und die Wahrscheinlichkeitsgrenze im Falle der Zeitsequenzanalyse lag bei 0,5. Es konnte eine Trefferquote des Profilnetzwerkes, wie es auch in Abbildung 5.3 dargestellt ist, von 70,1% bestimmt werden.
Dabei konnten die Auftrittswahrscheinlichkeiten aus Tabelle 5.3 für den illegalen Fall der benutzten Merkmale bestimmt werden.

Bezeichnung	Beschreibung	$P(illegal\|tripel)$
TRN_TYP	Kartenart	0,369
CURR_CD	Währung	0,377
POS_ENT_NT	Point Of Sale	0,208
ICA_CD	Transaktionsherkunft	0,190
SIC_CD	Branchenschlüssel	0,134
MSG_TYP	Nachrichtentyp	0,377
MER_ID	Händler-ID	0,120

Tabelle 5.3: Mißbrauchsauftrittswahrscheinlichkeiten verschiedener Merkmale

Die im Zusammenhang mit Mißbrauch bestimmten Auftrittswahrscheinlichkeiten machen deutlich, wie stark die einzelnen symbolischen Datentypen in die Klassenentscheidung der Zeitsequenzanalyse eingehen. Außerdem geht in den Wahrscheinlichkeitswert die Häufigkeit, mit der eine Wiederholung in Bezug zum Mißbrauch eintritt, mit ein. Die Werte entsprechen im Verhältnis den in Tabelle 5.1 global bestimmten Werten. Die Summe dieser Wahrscheinlichkeiten ist ausschlaggebend für die spezielle Klassenentscheidung.
Übertrifft diese die Schwelle von $0,5$, so konnten ausreichend viele Wiederholungen in den verschiedenen Datenfeldern registriert werden.

In der anschließenden Verifikationsphase konnte eine Trefferquote von 65,2% auf insgesamt 250 vom Training verschiedenen Datensätzen[9] erreicht werden. Die einzelnen Trefferquoten sind in Tabelle 5.4 aufgeführt. Die angegebene Konfidenz wurde auf Werten, die auf ein reales Verhältnis aus legalen und illegalen Daten hochgerechnet wurden, bestimmt (siehe auch Gleichung 4.23.

Fenster-größe	Treffer-quote	richtige Diagnose		falsche Diagnose		Konfidenz
		legal	illegal	legal	illegal	
3	65,2%	93,6%	36,8%	6,4%	63,2%	0,57%
4	58,0%	72,8%	43,2%	27,2%	56,8%	0,16%
5	62,0%	98,4%	25,6%	1,6%	74,4%	1,58%
6	59,0%	60,6%	57,5%	39,4%	42,6%	0,15%

Tabelle 5.4: Klassifizierungsergebnisse Profilauswertung

Bei den Auftrittswahrscheinlichkeiten der Wiederholungssequenzen im Falle der symbolischen Daten kommt es im Laufe der Fenstervergrößerung zu immer kleineren Werten bis hin zu einer Wahrscheinlichkeit von 0. Das heißt, das Auftreten immer längerer Wiederholungssequenzen ein und des selben Wertes wird immer unwahrscheinlicher.
Dies hat unter anderem auch zur Folge, daß die Gewichtungen der Analysemethoden ab einer Fenstergröße von 4 umschlagen; das heißt, war vorher die symbolische Profilauswertung für die letztendliche Klassenentscheidung verantwortlich, so ist es für die Fenstergrößen 5 und 6 die analoge Profilauswertung.

5.5.2 Interpretation und Auswertung der Ergebnisse

Der Verlauf der Trefferquoten beziehungsweise der Konfidenzergebnisse über die unterschiedlichen Fenstergrößen hinweg ist vorhersehbar. Mit der Verbreiterung des Diagnosefensters wird eine einheitliche Folge von gleichen Werten für die betrachteten symbolischen Merkmale immer unwahrscheinlicher. Die Summe der Auftrittswahrscheinlichkeiten erreicht seltener die vorgegebene Schwelle, so daß immer öfter und damit auch öfter irrtümlicherweise für eine legale Transaktion entschieden wird. Durch die in der Gesamtheit sehr unwahrscheinliche Folge von Werten, tritt dieses auf den symbolischen Daten basierende Entscheidungskriterium im Einfluß auf das abschließende, durch das lineare Neuron bestimmte, Ergebnis zurück. Es finden immer mehr Falschzuweisungen diesbezüglich statt. Stattdessen wird mehr Einfluß auf die Entscheidung auf Basis der analogen Daten gelegt. Damit sinkt folglich die Trefferquote seitens der legalen Datensätze, da weniger Entscheidungen nun zugunsten dieser Klasse getroffen werden. Eine Konfidenzverminderung ist die Folge. Desweiteren werden dadurch vermehrt illegale Datensätze richtig zugewiesen, da die oft falsche Entscheidung der symbolischen

[9] jeweils 125 Datensätze pro Klasse

Profilauswertung zugunsten einer legalen Transaktion durch die symbolische Auswertung wegfällt.
Zu bemerken ist in diesem Zusammenhang, daß die Gesamttrefferquote auf einem einheitlichen Niveau bleibt, sich also nicht sehr stark in Folge der Verschiebung der Entscheidungsgewichtung ändert.

Ein nicht zu unterschätzender Punkt bei der zeitabhängigen Transaktionsanalyse ist, daß sämtliche Transaktionsdaten nur eine Auswahl aller getätigten Transaktionen repräsentieren. Es liegen *nicht* alle, und damit sowohl illegale als auch legale Transaktionsdaten, des in B angegebenen Zeitintervall zur Analyse vor. Dadurch kann es trotz der zeitlichen Sortierung zu Zeitsprüngen bei dieser zeitabhängigen Auswertung kommen, die das Ergebnis verfälschen, und eine Klassifizierung auf den gegebenen Daten erschweren.
Auch hier wird deutlich, daß eine alleinige Auswertung der Daten mit Hilfe der Profilanalyse nicht ausschließlich für eine Mißbrauchsanalyse zur Mißbrauchsprävention geeignet ist, da die Zuordnungssicherheit in Form der Konfidenz nicht ausreichend ist.

Kapitel 6

Kombinierte Mißbrauchsanalyse mit sämtlichen Analysemethoden

6.1 Motivation

Nachdem nun in den Kapiteln 3 bis 5 verschiedene Modelle zur Analyse der unterschiedlichen Datentypen vorgestellt wurden, soll nun darauf aufbauend ein Modell zur Realisierung einer Mißbrauchserkennung und Prävention skizziert, hergeleitet und ausgewertet werden. Verschiedene Ansätze zur Umsetzung sind in diesem Zusammenhang denkbar, die zudem von der verfolgten Intention bei der Mißbrauchsanalyse abhängen. Es gilt, eine Entscheidung bezüglich des Analyseverhaltens zu treffen: Soll zum einen vermehrt Mißbrauch aufgedeckt werden, oder soll die Gewichtung mehr auf der Verminderung von Fehlalarmen ausgerichtet sein. Die vorgestellten Modellansätze bauen im wesentlichen auf den vorgestellten Methoden auf und unterscheiden sich in verschiedenen Eigenarten, angefangen bei der Architektur, der Abarbeitungsreihenfolge sowie Trainings- und Arbeitsgrundlagen der Auswertungen, was zu unterschiedlichen Ergebnissen bei der Einstufung der jeweiligen Transaktionen führt.

Man muß bei neuronalen Netzen oder allgemein bei adaptiven Verarbeitungsmethoden, wie sie nun erörtert werden, zwischen zwei Arbeitsmethoden unterscheiden: Zum einen benötigt das neuronale Netz eine Phase, in der Daten zum Training präsentiert und verarbeitet werden können, zum anderen wird in der Arbeitsphase die eigentliche Diagnose der Eingabedaten ausgeführt. Es ist durchaus denkbar, nachdem das Modell ausreichend angelernt wurde, auch das Training parallel zu der Diagnosephase auszuführen und damit die Diagnoseleistung ständig zu verbessern und an die aktuellen Gegebenheiten anzupassen. Eine Einschränkung ist jedoch durch die Regelgeneralisierung zu verzeichnen. Dieser Arbeitsschritt muß unabhängig von der Diagnose in regelmäßigen Abständen durchgeführt werden, um auf eventuelle Veränderungen der Mißbrauchseigenschaften

reagieren zu können. Anders verhält es sich bei den Regelstatistiken, die ebenfalls parallel zur Diagnose geführt werden können, wie es in 3 geschildert ist. Damit ändern sich dann im Verlauf der Regelfilterung unter Umständen die Konfidenz- und Abdeckungswerte, so daß bestimmte Regeln aus der Betrachtung herausfallen und andere hinzugenommen werden. Dennoch ist eine regelmäßige Generalisierung auf den aktuellen, statistischen Grundlagen sämtlicher Mißbrauchsdatensätze unerlässlich.

Zunächst wird einführend ein Modellansatz vorgestellt, der prinzipiell als die naheliegendste Realisierung angesehen werden kann. Dieser Ansatz basiert, wie auch die anderen vorgestellten Ansätze, auf den vorgestellten Analyseverfahren. Die einzelnen Analyseschritte werden sequentiell hintereinander ausgeführt, so daß es abschließend zu einer Transaktionszuordnung kommt. Bei diesem Ansatz werden die Datensätze entsprechend der Regeln, bei denen es im Falle der symbolischen Daten zu einer Übereinstimmung kommt, durch speziell abgestimmte und trainierte neuronale Netzwerke ausgewertet.
Ein zweiter Ansatz geht davon aus, daß unabhängig von der Regelfilterung eine allgemeine Analyse der Profil- und Analogdaten stattfindet. Die einzelnen Analyseschritte können somit parallel verlaufen und konkurrieren miteinander. Es kommt dabei je nach Gewichtung der einzelnen Zuordnungsergebnisse zu einer Dominanz der regelbasierten Entscheidung, so daß die binäre Ausgabe als eine Art Konjunktion sämtlicher Ergebnisse angesehen werden kann, die zum Endergebnis führt. Entweder entscheiden alle Analysemethoden gemeinsam, oder die Regelfilterung ist ausschlaggebend für das abschließende Ergebnis.
Im letzten Teil soll dieses Modell dahingehend optimiert werden, daß entsprechend der guten regelbaiserten Entscheidungen es wie im ersten Modell zu einer Staffelung der Analyseschritte kommt und somit eine Aussortierung der irrelevanten Daten durch den Regelfilter für die nachfolgenden Schritte erreicht wird. Dieses Verfahren baut auf einer hierarchischen Struktur auf, bei der die Filterung der Datensätze durch die generalisierten Regeln, ähnlich dem ersten Ansatz, im Vordergrund steht. Ergänzend soll anschließend diese auf ein Minimum reduzierte Datensatzmenge im Zweifelsfall von den weiteren Analyseverfahren, den Analogdaten verarbeitenden Netzwerken, eingehender untersucht und klassifiziert werden. Im Unterschied zum regelbasierten ersten Ansatz werden hier die Datensätze, die von allen generalisierten Regeln abgedeckt werden, zusätzlich durch ein Profil- und Analogdaten auswertendes Netz bearbeitet; es findet keine Spezialisierung bezüglich der einzelnen Abdeckungsregeln statt.

Durch die Unterscheidung der verschiedenen Modellansätze untergliedert sich dieses Kapitel wie folgt: Zunächst soll das Modell des intuitiven, regelspezifischen Ansatzes vorgestellt werden (Abschnitt 6.2). Die Ergebnisse dieses Modells werden in Bezug zu den in den einzelnen Kapiteln der unterschiedlichen Analyseverfahren vorgestellten Einzelergebnissen betrachtet. Im Anschluß daran wird das

parallele Klassifizierungsmodell vorgestellt (Abschnitt 6.3). Auch hier werden die Ergebnisse präsentiert und mit den bisher gesammelten verglichen und diskutiert. Aufgrund dieser Ergebnisse kann das hierarchische Modell hergeleitet werden (Abschnitt 6.4). Die mit diesem Modell erzielten Ergebnisse sollen anschließend vorgestellt und in Relation zu den Ergebnissen aus Abschnitt 6.3 diskutiert werden. Abschließend, in Abschnitt 6.5, werden die Vorzüge beziehungsweise die Nachteile der einzelnen Verfahren charakterisiert und gegeneinander abgewogen.

6.2 Ein regelspezifisches Modell

6.2.1 Motivation

Aufgrund der Tatsache, daß es sich bei mehr als der Hälfte der Daten eines Transaktionsdatensatzes um symbolische Werte handelt, soll in dem nun folgend beschriebenen *regelbasierten Modell* die Auswertung bezüglich dieser symbolischen Daten im Vordergrund stehen. Eine individuelle Klassifizierung der Datensätze, die von einer einzelnen Regel abgedeckt werden, führt zu einer sehr genauen Klassifizierung der Transaktionsdaten. Durch die Hinzunahme des in Abschnitt 5 beschriebenen Karteninhaberprofils kann jedoch diese Art der Auswertung auf Grund des begrenzten Datenmaterials auf den vorliegenden Daten nicht umgesetzt werden. Zudem kann der Aufwand in Relation zum Erfolg nicht gerechtfertigt werden, so daß ein auf einer größeren Regelmenge basierendes Analysemodell entworfen wurde, das im Abschnitt 6.4 eingehend beschrieben ist.

6.2.2 Das regelspezifische Analysemodell

Im Zuge der Generalisierung, wie sie in Kapitel 3 beschrieben ist, verringert sich die Konfidenz der einzelnen Regeln mit zunehmender Generalisierung, so daß eine eindeutige Assoziation mit den korrekten Mißbrauchstypen immer schwerer und ungenauer wird. Aus diesem Grunde wird versucht, die verlorene Konfidenz wieder anzuheben, das heißt, die Assoziation von Transaktion und Mißbrauchsart sicherer und genauer zu gestalten, indem zusätzlich das Ergebnis der Klassifizierung auf Basis der analogen Daten und der Profilauswertung zusätzlich zu der Entscheidungsfindung benutzt wird. Abbildung 6.1 zeigt einen möglichen Verlauf der Konfidenz einer Regel. Am Punkt `A` wurde die Schwelle der Mindestkonfidenz

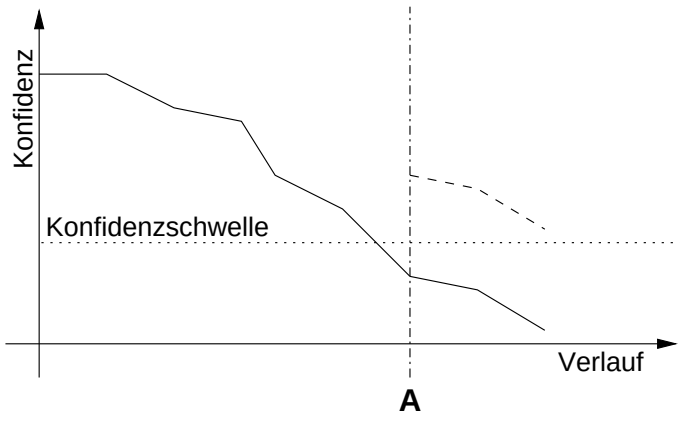


Abbildung 6.1: Möglicher Konfidenzverlauf

unterschritten. In diesem Fall setzt die Auswertung der analogen Daten ein und hilft, die Konfidenz der Regel wieder auf ein ausreichendes Maß anzuheben. Die

zusätzliche Klassifizierung mit Hilfe der Analogdaten wird durch das in Kapitel 4 vorgestellte RBF-Gesamtnetz realisiert. So wird ergänzend zu der Filterung durch die entsprechende generalisierte Regel der aktuelle Datensatz auf Basis der Analogdaten ausgewertet. Ein Beispiel soll dies verdeutlichen:

	x_a		x_b	x_c		x_d		N
generalisierte Mißbrauchsregel $\hat{R}_{ij}$	A	$\cdots$	$\star$	B	$\cdots$	$\star$	$\cdots$	$\star$
abgedeckte Mißbrauchsdatensätze	A	$\cdots$	C	B	$\cdots$	F	$\cdots$	-1
	A	$\cdots$	D	B	$\cdots$	G	$\cdots$	-1
	A	$\cdots$	E	B	$\cdots$	H	$\cdots$	$+1$
fälschlicherweise abgedeckte legale Datensätze	A	$\cdots$	U	B	$\cdots$	Z	$\cdots$	$+1$
	A	$\cdots$	V	B	$\cdots$	Y	$\cdots$	-1
	A	$\cdots$	W	B	$\cdots$	X	$\cdots$	$+1$

Abbildung 6.2: Regelfilterung und Klassifizierung

Aus der Menge der zu analysierenden Transaktionsdaten werden genau die herausgefiltert, die übereinstimmend mit der generalisierten Regel $\hat{R}_{ij}$ sind, mit $j > 0$. In Abbildung 6.2 sind das 6 Datensätze, die mit der generalisierten Mißbrauchsregel an den Stellen x_a und x_c übereinstimmen. Diese generalisierte Regel enthält die spezifischen Werte x_a = A und x_c = B. An anderen Stellen symbolischer Datentypen x_b und x_d wurde durch die Generalisierung ein Wildcard ($\star$) gesetzt. Auch die hinzugefügte Stelle N, die später die zusätzliche Klassifizierungsentscheidung enthalten wird, ist zunächst mit einem solchen Platzhalter markiert.
Durch die Einstufung durch vorher trainierte, neuronale Netze, kann dann eine unterstützende Entscheidung, angefügt im Feld N, bezüglich der Klassenzuordnung des Datensatzes getroffen werden. Dabei entspricht der Wert -1 einer illegalen Transaktion, der Wert $+1$ einer legalen Transaktion. Die Konfidenz c bezüglich des Mißbrauchs wächst damit im Beispiel von vorher $c = \frac{3}{3+3} = 0,5$ auf $c' = \frac{2}{2+1} = 0,67$, was einer Steigerung von 33% entspricht.

In diesem Zusammenhang sind verschiedene Ansätze denkbar. Diese basieren unter anderem auf der Tatsache, daß das Training für die spätere Klassifizierung von ausschlaggebender Bedeutung ist. So können verschiedene Trainingsszenarien zu unterschiedlichen Klassifizierungsergebnissen führen.
Eine Möglichkeit, das Training unterschiedlich zu gestalten, setzt bei den Datensätzen an, die zum Training verwendet werden sollen. Zum einen können die beteiligten RBF-Netze auf dem von den generalisierten Regeln eingeschränkten Datenraum trainiert werden, zum anderen auf *allen* vorhandenen Datensätzen.
Weiter können auch die Trainingsläufe an das Verhältnis aus legalen und illegalen Datensätzen angeglichen werden, so daß nicht paritätisch, das heißt gleichverteilt, Datensätze aus dem Datenraum rekrutiert, sondern in ungleichem Verhältnis

mehr legale Daten berücksichtigt werden, wie es schon in Abschnitt 4.5.3 vorgestellt wurde.
Außerdem ist die Art und Weise, mit der die Datensätze abschließend bei der Klassifizierung untersucht werden, ebenfalls ausschlaggebend. So kann erneut entweder auf den gesamten Daten oder auf dem von den generalisierten Regeln eingeschränkten Datenraum eine Klassifizierung durchgeführt werden. Es werden also folgende Trainingsszenarien auf den anschließenden Klassifizierungserfolg hin untersucht:

1. (a) Die Netzschicht wurde auf **allen** Datensätzen trainiert. Dabei wurden jedoch zur Verifikation nur die von den generalisierten Regeln **eingeschränkten** Datensätze klassifiziert.

   (b) Die Netzschicht wurde **speziell** auf den Datensätzen trainiert, die von einer generalisierten Regel abgedeckt wurden. Die Klassifizierung wurde dabei wiederum auf derart **eingeschränkten** Regeln verifiziert.

2. (a) Die Netzschicht wurde auf **allen** Datensätzen trainiert, und anschließend wurde auch die Verifikation auf **allen** Daten ausgeführt. Diese Variante soll erneut die Notwendigkeit der vorangegangenen Generalisierung und der Einschränkung durch diese Regeln verdeutlichen.

   (b) Die Netzschicht wurde zwar **speziell** auf den Datensätzen trainiert, die von einer generalisierten Regel abgedeckt wurden, jedoch wurde anschließend zur Verifikation versucht, **alle** Datensätze mit diesem Netz in die richtigen Klassen einzuordnen.

Der Fall 2a) entspricht dem allgemeinen Training auf den Rohdaten, wie es schon in Abschnitt 4.5.3 vorgestellt wurde. Dieses Szenario berücksichtig in keiner Weise die Generalisierung, soll aber an dieser Stelle als Vergleich für die weiter ermittelten Ergebnisse dienen. Es kann damit die Notwendigkeit der Auswertung durch die generalisierten Regeln bestätigt und hervorgehoben werden. Fall 2b) wurde nur der Vollständigkeit halber aufgeführt, soll aber nicht weiter betrachtet werden, da keine Verbesserung bezüglich der Klassifizierungen zu erwarten ist.
Die Fälle 1b) und 1b) verfolgen am ehesten die Intention der Generalisierung. Hauptziel der generalisierten Regeln ist es, den Transaktionsraum derart einzuschränken, daß eine zusätzliche Klassifizierung spezialisiert und erfolgreich bewerkstelligt werden kann. Auf diese Weise wird nun theoretisch jeder generalisierten Regel, beziehungsweise den damit abgedeckten Transaktionsdatensätzen, ein „eigenes Netz“ zur Klassifizierung der Analogdaten mitgegeben und antrainiert. Natürlich werden an dieser Stelle nur die Gewichte und andere notwendige Parameter des Netzes mit der entsprechenden Regel in Verbindung gebracht.

Im folgenden Abschnitt 6.2.4 sind nun die diversen Testläufe anhand einiger Beispiele beschrieben und ausgewertet. Es stellt sich jedoch an dieser Stelle die Frage

nach der Berechnung der durch die Analogdatenanalyse ergänzten Konfidenzwerte. Dazu läßt sich folgender Satz herleiten:

Satz 3 *Die kombinierte Konfidenz c_{combi} für die Einordnung durch die generalisierten Regeln unter Hinzunahme der trainierten neuronalen Netze läßt sich mit Hilfe der entsprechenden Trefferquoten bei der Klassifizierung berechnen. Dabei kann auf die bei der Regelfilterung bestimmte Konfidenz c_{regel} zurückgegriffen werden.*

Beweis 3 *Zunächst sind die bekannten Tatsachen zusammenzufassen. So gilt für die durch die generalisierten Regeln gegebene Konfidenz c_{regel} entsprechend Gleichung 3.17:*

$$\begin{aligned} c_{regel} &= \frac{m_{regel}}{m_{regel} + l_{regel}} \\ &\textit{mit} \quad m_{regel} = \|\{M_{>0}|\mathbf{x}\}\| \quad \textit{und} \quad l_{regel} = \|\{M_0|\mathbf{x}\}\| \end{aligned} \tag{6.1}$$

Für die kombinierte Konfidenz gelte:

$$\begin{aligned} c_{combi} &= \frac{m_{combi}}{m_{combi} + l_{combi}} \\ &\textit{mit} \quad m_{combi} = \|\{(M_{>0}|\mathbf{x}) \wedge (N = -1)\}\| \\ &\textit{und} \quad l_{combi} = \|\{(M_0|\mathbf{x}) \wedge (N = -1)\}\| \end{aligned} \tag{6.2}$$

Für die Einordnung durch die neuronalen Netze seien folgende Trefferquoten für die Berechnung der kombinierten Mißbrauchskonfidenz c_{combi} von Bedeutung:

richtige Mißbrauchsprognose: *Es kommt zu einer berechtigten Mißbrauchsentscheidung durch das trainierte neuronale Netz. Es gilt formal:*

$$p_t^{NN} = P\big((M_{>0}|\mathbf{x}) \wedge (N = -1)\big) = \frac{\|\{(M_{>0}|\mathbf{x}) \wedge (N = -1)\}\|}{\|\{M_{>0}|\mathbf{x}\}\|} = \frac{m_{combi}}{m_{regel}}$$

falsche Mißbrauchsprognose: *Das trainierte RBF-Netzmodell weist einem legalen Datensatz fälschlicherweise einen Mißbrauch zu. Dies kann formal ausgedrückt werden als:*

$$p_f^{NN} = P\big((M_0|\mathbf{x}) \wedge (N = -1)\big) = \frac{\|\{(M_0|\mathbf{x}) \wedge (N = -1)\}\|}{\|\{M_0|\mathbf{x}\}\|} = \frac{l_{combi}}{l_{regel}}$$

Damit läßt sich nun die Konfidenz der kombinierten Analyse c_{combi} durch Regelfilterung und Analogdatenauswertung mit dem neuronalen Netz durch Auflösen nach m_{combi} und l_{combi} sowie Einsetzen in 6.2 wie folgt bilden:

$$c_{combi} = \frac{m_{regel} \cdot p_t^{NN}}{m_{regel} \cdot p_t^{NN} + l_{regel} \cdot p_f^{NN}} \tag{6.3}$$

Löst man Gleichung 6.1 nach l_{regel} auf, so erhält man weiter

$$l_{regel} = m_{regel} \cdot \left(\frac{1}{c_{regel}} - 1 \right)$$

und somit

$$c_{combi} = \frac{p_t^{NN}}{p_t^{NN} + p_f^{NN} \cdot \left(\frac{1}{c_{regel}} - 1 \right)} \tag{6.4}$$

□

Damit ist eine Gleichung gegeben, mit der auf Basis der Regelkonfidenz und der Klassifizierungswahrscheinlichkeiten der Analogdaten und Profildaten auswertenden Netzmodellen die Gesamtkonfidenz berechnet werden kann.
Weiter muß bei der Berechnung der Gesamtkonfidenz beachtet werden, daß der Datenumfang, entsprechend den Absolutzahlen m_{regel} und l_{regel}, durch die Regelfilterung eingeschränkt ist und diese Werte wie in Gleichung 6.3 angegeben, verwendet werden.

6.2.3 Regelgrundlage

Zu Simulationszwecken wurde ein Generalisierungslevel ausgewählt, und die enthaltenen Regeln für die Realisierung dieses Ansatzes entsprechend ausgewertet. Je mehr Wildcards in den Regeln enthalten sind, um so mehr Transaktionsdaten werden abgedeckt und stehen für die Auswertung durch das analogdatenverarbeitende Netz zur Verfügung. Darum sollen stellvertretend an dieser Stelle zwei Regeln des GL 16 betrachtet und die Vorgehensweisen anhand dieser Regeln demonstriert werden.
Zunächst jedoch sollen die Regeln des 16. Generalisierungslevels näher beschrieben werden. Es handelt sich bei den in Tabelle 6.1 abgebildeten Regeln um sehr weit generalisierte Vertreter der Mißbrauchsdatensätze bezüglich der symbolischen Daten. Sie besitzen entsprechend des fortgeschrittenen Generalisierungslevels insgesamt 16 Wildcards, die jedoch an unterschiedlichen Stellen platziert sind.
Die Regeln dieses Generalisierungslevels konnten während der Generalisierung die Konfidenz und Abdeckungswerte aus Tabelle 6.2 erzielen. Auf sämtlichen vorliegenden Daten ändert sich das Verhältnis entsprechend Tabelle 6.3. Hier wurde die Anzahl der von den Regeln abgedeckten, legalen Datensätze unter anderem auf das tatsächliche Verhältnis von 1:1000 aus illegalen und legalen Daten hochgerechnet. Da 5850 Mißbrauchsdatensätze vorliegen, wurden die Werte für die legalen Datensätze auf den Umfang von 5850000 Transaktionen geschätzt. Dies ergibt bei 542858 legalen Datensätzen einen Hochrechnungsfaktor von 10,776.

Regel	ACCT_NBR	TRN_TYP	CURR_CD	POS_ENT_CD	FAL_SCOR	CRD_TYP	ICA_CD	AID_CD	SIC_CD	ACT_CD
1	⋆	EA	840	⋆	⋆	EM	2768	8403184	⋆	000
2	⋆	EA	840	ZZUTSZ1UZZZ1	0	EM	⋆	⋆	563	000
3	⋆	EA	840	⋆	0	EM	2768	8403184	⋆	⋆
4	⋆	EA	840	⋆	995	EM	⋆	⋆	⋆	000

Regel	MSG_TYP	MER_ID	MER_CNTY_CD	CTY_1	POST_CD_1	CNTY_CD_1	CR_LMT	ACTV_IND	ACCT_STAT	CTY_2	POST_CD_2	ADDR_STAT	EMIT_NBR	INST_NBR	ISS_REAS	GEN_CD	CARD_TYP
1	1100	⋆	000	⋆	⋆	⋆	⋆	I	⋆	⋆	⋆	null	⋆	⋆	N	⋆	⋆
2	1100	⋆	000	⋆	⋆	⋆	⋆	I	⋆	⋆	⋆	null	⋆	⋆	⋆	⋆	⋆
3	1100	⋆	000	⋆	⋆	⋆	⋆	I	⋆	⋆	⋆	null	⋆	⋆	⋆	002	⋆
4	1100	⋆	000	⋆	null	⋆	⋆	I	F8	⋆	⋆	null	⋆	⋆	⋆	⋆	⋆

Tabelle 6.1: Regeln des Generalisierungslevel 16

Regel	Konfidenz	Abdeckung	Support	# illegal	# legal
1	0,21	0,12	0,011	690	28
2	0,22	0,01	0,001	78	3
3	0,32	0,05	0,004	267	6
4	1,00	0,01	0,001	42	0

Tabelle 6.2: Regelwerte des GL 16 bei Generalisierung auf Teilmengen

Dabei zeigt sich, daß nur die 1. und die 3. Regel ausreichend legale Datensätze auf den vorliegenden Daten abdecken, so daß eine weitere Verarbeitung nur mit diesen Regeln möglich ist.
Es ist unter diesen Umständen sogar denkbar, weiter generalisierte Regeln auszuwählen, die zwar als Preis eine niedrigere Konfidenz aufweisen, dafür aber mehr die Möglichkeit bieten, die speziell abgedeckten Datensätze in einem größeren Umfang untersuchen zu können. Die Abdeckung weiter generalisierter Regeln ist sowohl für den legalen wie auch für den illegalen Fall entsprechend höher, und es stehen vermehrt Datensätze zur Analogdatenauswertung zur Verfügung. Dies würde jedoch einen fortgesetzt oder erneuten Generalisierungsdurchgang mit veränderten Parametern erfordern. Für eine Simulation sollen jedoch die beiden Regeln 1 und 3 genügen.

Regel	# in Teilmengen		# auf allen	hochgerechnete #	Konfidenz
	illegal	legal	legalen Daten	auf legalen Datensätzen	
1	690	28	500	5388,000	11,352%
2	78	3	47	506,472	13,345%
3	267	6	64	689,664	27,910%
4	42	0	0	0,000	100,000%

Tabelle 6.3: Auftrittshäufigkeiten der Regeln

6.2.4 Klassifizierungsergebnisse

Es werden nun die Ergebnisse der entsprechenden Trainingsverfahren vorgestellt, wie auch die Resultate der anschließenden Verifikations- oder Arbeitsphase. Bei letzterem wird versucht, so weit es die zugrundeliegende Daten und deren Umfang zulassen, eine zu den beim Training verwendeten Datensätzen disjunkte Menge zu verwenden.

Allgemeines, paritätisches Training

Zunächst wurde auf der allgemeinen Datenmenge ein Training mit 300 Trainingszyklen, also mit jeweils 150 Daten aus den legalen sowie illegalen Daten durchgeführt. Die Trefferquote liegt bei ca. 82,3%, wie auch schon in Abschnitt 4.5 gezeigt werden konnte.
Der anschließende Verifikationslauf wurde auf verschiedenen Datenmengen durchgeführt, zunächst jedoch auf 200 beliebigen Daten, also *ohne* Einschränkung durch eine generalisierte Regel. Diese Analyse entspricht im Prinzip der Auswertung in Abschnitt 4.5.2, jedoch ist es dadurch möglich, diese Ergebnisse mit den auf Basis der eingeschränkten Rohdaten erzielten, direkt vergleichen zu können. In Tabelle 6.4 sind die Ergebnisse der allgemeinen Auswertung mit dem Symbol $\forall$ bezeichnet. Die Konfidenz berechnet sich in diesem Fall in Anlehnung an Gleichung 3.17 auf Seite 31, wobei allerdings die Fehlalarmerate in Form von mit Mißbrauch markierten, legalen Datensätzen um den Faktor 1000 hochgerechnet werden muß.
Desweiteren wurde die Klassifizierung auf den von den einzelnen Regeln aus GL 16 eingeschränkten Datensätzen ausgeführt. Die Trefferquoten sind ebenfalls in Tabelle 6.4 zusammengefaßt dargestellt. Außerdem wurde die Konfidenz zu den erhobenen Werten wie in folgender Gleichung entsprechend der Konfidenzgleichung 6.3 aus Satz 3 berechnet.

$$confidence_{Regel\ 1} = \frac{690 \cdot 0,8}{690 \cdot 0,8 + 5388 \cdot 0,09} = 0,53235 = 53,235\%$$

Dabei ist zu beachten, daß sich die Konfidenz in diesem Fall *nur* auf die von der Regel (eventuell hochgerechnete) Anzahl an abgedeckten Regeln bezieht, wie in der Berechnung zu sehen ist.

Regel	# Daten/ Klasse	richtig		falsch		Treffer- quote	Konfi- denz
		legal	illegal	legal	illegal		
∀	100	87,0%(87)	73,0%(73)	13,0%(13)	27,0%(27)	80,0%	0,6%
1	100	91,0%(91)	80,0%(80)	9,0%(9)	20,0%(20)	85,5%	53,2%
3	64	100,0%(64)	73,4%(47)	0,0%(0)	26,6%(17)	86,7%	100,0%

Tabelle 6.4: Klassifizierungsergebnisse in Folge von allgemeinem, paritätischem Training

Wie festzustellen ist, können im Falle der Klassifizierung auf den eingeschränkten Datensätzen bessere Konfidenzwerte als bei der Abdeckung alleine durch die Regeln an sich festgestellt werden. Aus diesem Grund soll der Zugewinn im Vergleich zu der einfachen Regelfilterung auf Basis der generalisierten Regeln explizit wie in Tabelle 6.5 angegeben werden.

Regel	Konfidenz Regelfilter	Konfidenz aktuell	Verbesserung
1	11,35%	53,24%	369%
3	27,91%	100,0%	258%

Tabelle 6.5: Verbesserung durch zusätzliches, paritätisch trainiertes Netz auf allen Daten

Spezielles, paritätisches Training

In diesem Abschnitt soll nun der Zugewinn bei der Klassifizierung gezeigt werden, der erreicht werden kann, wenn auf den von den generalisierten Regeln eingeschränkten Daten trainiert wird. Dazu wird das Netz zunächst mit einer begrenzten Anzahl an gefilterten Trainingsdaten trainiert und anschließend zur Klassifizierung auf ebenfalls eingeschränkten Datensätzen herangezogen. Eine Klassifizierung der Rohdaten mit derart trainierten Netzen ist belanglos, da kein besseres Zuordnungsverhalten aufgrund des eingeschränkten Trainings zu erwarten ist als im vorhergehenden Abschnitt, des allgemeinen, paritätischen Trainings. Die Ergebnisse der Klassifizierung sind in Tabelle 6.6 zusammengefaßt. Zum Training auf den Daten, die von Regel 1 aus GL 16 abgedeckt werden, wurden zunächst insgesamt wieder 300 Daten verwendet. Um die gleiche Anzahl an „unbenutzten“ Daten zur Verifikation zur Verfügung zu haben, wurde für Regel 2 jedoch nur ein Training auf insgesamt 64 Daten durchgeführt, um dieselbe Anzahl an nicht zum Training verwendeten Daten auch zur Verifikation vorrätig zu haben.

Erneut ist auch hier eine Konfidenzsteigerung wie auch eine Verbesserung der Trefferquoten zu verzeichnen:

Regel	# Daten/ Klasse	richtig		falsch		Treffer- quote	Konfi- denz
		legal	illegal	legal	illegal		
1	100	96,0%(96)	81,0%(81)	4,0%(4)	19,0%(19)	88,5%	72,2%
3	32	87,5%(28)	62,5%(20)	12,5%(4)	37,5%(12)	75,0%	74,9%

Tabelle 6.6: Klassifizierungsergebnisse in Folge von speziellem, paritätischem Training

Regel	Konfidenz Regelfilter	Konfidenz aktuell	Verbesserung
1	11,35%	72,17%	536%
3	27,91%	74,87%	168%

Tabelle 6.7: Verbesserung durch zusätzliches, paritätisch trainiertes Netz auf regelspezifischen Daten

Allgemeines, ungleichmäßiges Training

Wie schon in Abschnitt 4.5.3 erörtert, können auch an dieser Stelle Auswirkungen des ungleichmäßigen Trainings auf die Trefferquote beziehungsweise auf die Konfidenz verzeichnet werden. Dazu werden die Netzwerkparameter mit Trainingsdaten in unterschiedlichen Verhältniskombinationen trainiert. Auf diese Weise kann auch das Training an das reale Verhältnis aus legalen und illegalen Daten angeglichen werden. Die anschließende Klassifizierung kann jedoch wieder wechselseitig, auf legalen und illegalen Daten gleichermaßen, durchgeführt werden, also im Verhältnis 1:1.

In Folge der Ergebnisse im Abschnitt 4.5.3 soll ein Trainingsverhältnis von 3:1 angenommen werden, da hier eine akzeptable Trefferquote seitens der illegalen Daten erzielt werden konnte. Es werden insgesamt 300 Trainingszyklen durchgeführt. Das heißt genau, daß 225 legale Daten und 75 illegale Daten dem Netz zum Training bereitgestellt werden. Anschließend werden die regelspezifischen Datensätze mit dem so trainierten Netz paritätisch klassifiziert. Die Trefferquote beim Training liegt erneut im Bereich von 80% (siehe Abschnitt 4.5.3). Die Ergebnisse der Klassifizierung sind in Tabelle 6.8 zusammengefaßt dargestellt.

Regel	# Daten/ Klasse	richtig		falsch		Treffer- quote	Konfi- denz
		legal	illegal	legal	illegal		
∀	100	98,0%(98)	60,0%(60)	2,0%(2)	40,0%(40)	79%	2,9%
1	100	100,0%(100)	73,0%(73)	0,0%(0)	27,0%(27)	86,5%	100,0%
3	64	98,4%(63)	73,4%(47)	1,6%(1)	26,6%(17)	85,9%	99,5%

Tabelle 6.8: Klassifizierungsergebnisse in Folge von allgemeinem, ungleichmäßigem Training (3:1)

Die Ergebnisse zeigen, soweit möglich, den erwarteten Konfidenzgewinn durch ein ungleichmäßiges Training des Netzmodells. Die Klassifizierung auf den von den Regeln abgedeckten Datensätzen erreicht allein durch ein unparitätisches Training eine 100%ige Konfidenz. Zwar ist ein Ansteigen der legalen Trefferquoten im Vergleich zu den Ergebnissen beim allgemeinen, paritätischen Training zu verzeichnen, doch rutschen die illegalen Trefferquoten auf niedrigere Werte ab. Die Gesamttrefferquote bleibt durchgehend auf dem gleichen Niveau. Die Konfidenz erreicht damit für die von den Regeln abgedeckten Datensätzen einen nahezu maximalen Wert. Die Konfidenz bezüglich der Klassifizierung auf allen Daten konnte, wie schon im Abschnitt 4.5, durch das unausgewogene Training nur wenig angehoben werden; in diesem Zusammenhang übt der Hochrechnungsfaktor von 1000 einen zu starken Einfluß auf die Konfidenzberechnung aus.

Regel	Konfidenz Regelfilter	Konfidenz aktuell	Verbesserung
1	11,35%	100,0%	781%
3	27,91%	99,48%	256%

Tabelle 6.9: Verbesserung durch zusätzliches trainiertes Netz im Verhältnis 3:1 auf allen Daten

Spezielles, ungleichmäßiges Training

Nun sollen abschließend die Auswirkungen eines ungleichmäßigen Trainings erprobt werden, wenn zusätzlich ausschließlich auf von den Regeln gefilterten Datensätzen trainiert wird. Aufgrund der nur sehr kleinen Datenmengen von maximal 64 legalen Daten im Falle von Regel 3 soll in diesem Fall ausnahmsweise Training und Verifikation auf den gleichen Daten stattfinden. Dies kann zu wesentlich besseren Ergebnissen im Vergleich zu den vorangegangenen Testläufen führen, da in diesem Fall die Neurone genau auf die Daten ausgerichtet und trainiert sind, die es zu erkennen gilt. Das Trainingsverhältnis soll erneut 3:1 betragen, um einen Vergleich zu den vorhergegangenen Ergebnissen zu ermöglichen. Die Ergebnisse sind in Tabelle 6.10 aufgeführt.

Regel	# Daten/ Klasse	richtig		falsch		Treffer- quote	Konfi- denz
		legal	illegal	legal	illegal		
1	100	99,0%(99)	82,0%(82)	1,0%(1)	18,0%(18)	90,5%	91,3%
3	64	100,0%(64)	73,4%(47)	0,0%(0)	26,6%(17)	86,7%	100,0%

Tabelle 6.10: Klassifizierungsergebnisse in Folge von speziellem Training im Verhältnis 3:1

Im Training der 1. Regel werden wieder unter den gegebenen Umständen 225 legale und 75 illegale Datensätze zum Training verwendet; bei Regel 3 sind es

64 legale und 21 illegale, also insgesamt 85 Datensätze. Die Trefferquote beim Training zu Regel 1 beträgt 85,3%, bei Regel 3 sind es 88,2%.
Zunächst sind die guten Trefferquoten seitens der legalen Daten zu erwähnen. Es entstammen 3/4 der Trainingsdaten aus dem legalen Datenbereich, so daß eine wesentlich bessere Zuordnungssicherheit bei der verifizierenden Zuordnung dieser Daten gegeben ist. Der Konfidenzgewinn ist in diesem Zusammenhang nahezu maximal. Der Zugewinn soll auch hier in Relation zu den durch die Regeln erreichten Konfidenzwerte gesetzt werden:

Regel	Konfidenz Regelfilter	Konfidenz aktuell	Verbesserung
1	11,35%	91,305%	704,3%
3	27,91%	100,0%	258,3%

Tabelle 6.11: Verbesserung durch zusätzliches trainiertes Netz im Verhältnis 3:1 auf von den Regeln eingeschränkten Daten

Man erkennt weiterhin, daß die Mißbrauchstrefferquoten sich aufgrund des mangelnden Trainings nicht verschlechtert haben. Aber auch eine Verbesserung ist nicht zu registrieren, die im Falle von Regel 3 durch die Verwendung der gleichen Daten für Training und Verifikation zu erwarten war.

6.2.5 Zusammenfassung und Auswertung

Die Ergebnisse dieses Modellansatzes haben gezeigt, daß mit Hilfe der neuronalen Netze die Konfidenzwerte der einzelnen Regeln in einem hohen Maße angehoben werden können. Es können Konfidenzsteigerungen in Bezug zur Regelkonfidenz um weit über das Hundertfache erreicht werden. Bei weiteren, jedoch undokumentierten Testreihen konnten ebenfalls auf größeren Datenmengen im Durchschnitt Konfidenzwerte von mehr als 50% mit der Kombination aus unparitätischem Training auf den gefilterten Daten erreicht werden.

Bei der Auswertung darf jedoch nicht außeracht gelassen werden, daß mit der Zielsetzung zur Erhöhung der Konfidenz besonders die Trefferquote bezüglich der illegalen Zuweisungen vernachlässigt wird. Auf die Konfidenz wirkt sich hauptsächlich die legale Trefferquote aufgrund des Hochrechnungsfaktors aus. Vergleicht man die Gesamttrefferquoten, so erkennt man generell einen logischen Anstieg mit der Spezialisierung auf den von einer Regel abgedeckten Datensätzen (Tabelle 6.4 und 6.6). Dieser Anstieg wird durch eine verbesserte Trefferquote seitens der legalen oder der illegalen Zuweisungen erreicht. Der eingeschränkte Datenraum ermöglicht es demzufolge, eine Unterteilung in die Klassen mit Hilfe der neuronalen Netze einfacher und besser gestalten zu können. Eine Anhebung des Trainingsverhältnisses zugunsten der legalen Daten zeigt in diesem Zusammenhang

nur wenig Auswirkung. Die Trefferquoten bei den legalen Zuweisungen können nicht viel weiter optimiert werden, sie stoßen schon bei dem paritätischen Training nahezu an die Grenzen des Möglichen. Für die Erfolgsquote seitens der illegalen Datensätze bringt ein solches unparitätisches Training eher nur Nachteile mit sich, da oft zu wenige Mißbrauchstransaktionen dem Netz für die anschließende Klassifizierung präsentiert werden.
In Zusammenhang mit der Mißbrauchsprävention bedeuten die erreichten Eregebnisse, daß die Entscheidung bezüglich des Mißbrauchs wieder auf ein sicheres Niveau angehoben werden kann. Die Fehlalarmrate sinkt mit dieser Konfidenzsteigerung, die Klassifikationssicherheit bezüglich des Mißbrauchs nimmt zu. Die Mißbrauchserkennung im Vergleich zu der allgemeinen Auswertung auf allen Daten kann jedoch unter Umständen durch eine zusätzliche Regeleinschränkung verbessert werden, wie man an den Tabellen 6.6 und 6.10 erkennt. Hier werden generell bessere Werte bei den illegalen Trefferquoten erzielt als bei den allgemein trainierten Netzauswertungen.

6.3 Das allgemeine Klassifikationsmodell

6.3.1 Motivation

In diesem Ansatz werden die einzelnen Analysemodelle gleichberechtigt nebeneinander ausgewertet und konkurrierend eingesetzt. Dabei wird die Einflußnahme jedes einzelnen Analysemodells im Laufe des Trainings gewichtet, so daß abschließend eine auf Basis einer Schwellenwertoperation ausgeführte binäre Klassifizierung stattfinden kann. Ziel dabei ist es, mit Hilfe dieser Architektur möglichst hohe Trefferquoten auf beiden Seiten der Datenzuordnung zu erzielen. Zwar steht die Trefferquote bezüglich der *illegalen* Daten bei der Mißbrauchsprävention im Vordergrund, jedoch ist es ebenso von entscheidender Bedeutung, daß besonders die Klassenentscheidung in Bezug zu den *legalen* Datensätzen möglichst fehlerfrei gestaltet wird, da es sich ja bei dem Gros der in der Realität vorkommenden Daten um eben jene legalen Autorisierungsanfragen handelt. Als Vergleichsmaß steht hier erneut die Konfidenz im Vordergrund. Mit der Optimierung der legalen Trefferquote sinkt das Risiko von Fehlalarmen, die es aus Kostengründen und besserem Kundenservice zu vermeiden gilt.

6.3.2 Modellbeschreibung

Dieses Modell faßt die in den Kapiteln 3 bis 5 vorgestellten verschiedenen Modelle in einer Entscheidung zusammen. Das heißt, die Ergebnisse der einzelnen Arbeitsschritte können parallel ausgewertet werden. Abbildung 6.3 skizziert in groben Zügen die Reihenfolge der Abarbeitung und die Zusammenhänge. Im wesentlichen handelt es sich bei diesem Modell um eine einfache Aneinanderreihung der bisher beschriebenen Analysemodelle. Dabei kommt es jedoch technisch gesehen nicht auf die Reihenfolge der Arbeitsschritte an, da abschließend die Einzelergebnisse gewichtet und zu einer Klassenentscheidung verrechnet werden. Diese letztendliche Auswertung der Binärausgaben der einzelnen Analysemethoden wird durch ein lineares Neuron, entsprechend dem in Abschnitt 4.2.3 vorgestellten, ausgeführt.

6.3.3 Teil- und Zwischenergebnisse

Im folgenden werden nun zusammenfassend die einzelnen Schritte mit den erreichten Ergebnissen vorgestellt und miteinander verglichen. Dabei soll nicht weiter auf die Einzelergebnisse der verschiedenen Analysemethoden eingegangen werden, da diese schon in den jeweiligen Kapiteln, in denen die einzelnen Verfahren vorgestellt wurden, erörtert und diskutiert wurden.

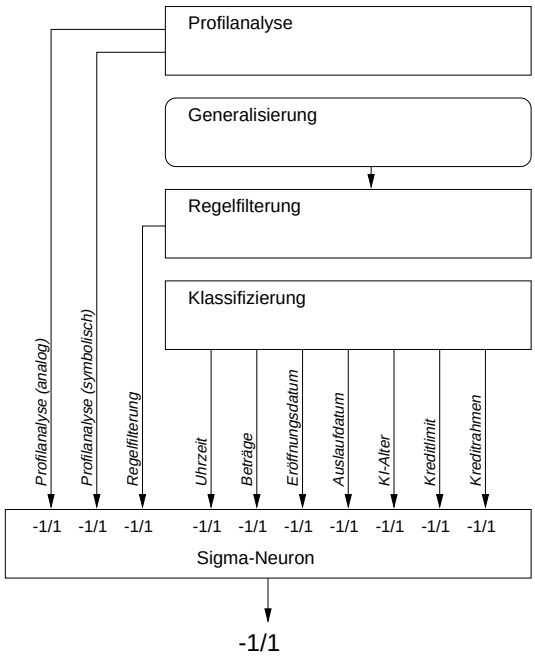


Abbildung 6.3: Klassifikationsmodell

Profilanalyse

Die Profilauswertung ist aus praktischen Gründen an den Anfang der Abarbeitung getreten, da nur Transaktionen *der* Konten verwendet werden können, die insgesamt ausreichend Buchungen getätigt haben. Dies ist abhängig von der Fenstergröße, auf der die jeweilige Profilauswertung aufbaut[1]. Für die begrenzte Datenmenge, die zu Testzwecken vorliegt, ist dies als einschränkender Faktor zu werten. Eine Alternative ist, die fehlenden Transaktionen durch Standarddatensätze zu ersetzen; jedoch würde damit das Bild des Profils stark verfälscht. Der Algorithmus sucht also nach Konten mit ausreichend Autorisierungsanfragen und wertet sie entsprechend den in Kapitel 5 vorgestellten Verfahren aus. In der Realität kann es zu dieser Diskrepanz nicht kommen, da spätestens nach einer konstanten Transaktionsanzahl eine ausreichende legale Transaktionshistorie für jedes Konto existiert. Eine kontospezifische Historie, die bei jeder Transaktion übermittelt wird, könnte hier zu einer effizienten Auswertung beitragen und langwierige Datenbankanfragen erübrigen.
Die Ausgaben der zur Profilanalyse herangezogenen Netz- beziehungsweise Analysemethoden werden jeweils einzeln in der Datenbank an die jeweiligen Transaktionen angehängt. Dies erlaubt auch später eine eigene Gewichtung der unterschiedlichen Auswertungsmethoden. Auf diese Weise kann das Training sowie auch der Arbeitsschritt unabhängig vom eigentlichen, gesamten Analyseverfahren getrennt ausgeführt und unter Umständen individuell aktualisiert und angewandt

[1] siehe dazu auch Abschnitt 5.1.2

werden.
Die derart gekennzeichneten Transaktionen dienen nun den weiteren Verarbeitungsschritten, wie sie auch im Flußdiagramm 6.3 abgebildet sind, als Analysedaten. Auf diese Weise ist gewährleistet, daß für die gesamte Mißbrauchsanalyse, bestehend aus den einzelnen Analyseschritten, dieselben Datensätze untersucht werden. Auf diese Weise ist ein abgesicherter Vergleich der Ergebnisse möglich.

Wie in Abschnitt 5.5 wird nun die Profilanalyse auf 1000 Datensätzen durchgeführt. Entsprechend kann auch hier eine Gesamttrefferquote von 68,5% erreicht werden. Es läßt sich unter den gegebenen Umständen erneut ein nur sehr geringer Konfidenzwert von 0,656% berechnen, da die Fehlerquote seitens der legalen Transaktionen zu hoch, aufgrund des Hochrechungsfaktors für ein reales Verhältnis von 1:1000, ausfällt. Im übrigen ist die Erfolgsquote bezüglich der illegalen Datensätze nur sehr gering.

Diagnose	legal	illegal	gesamt	Konfidenz
richtig	93,4% (467)	43,6% (218)	**68,5%**	0,656%
falsch	6,6% (33)	56,4% (282)	31,5%	

Tabelle 6.12: Profilanalyse auf 1000 Datensätzen

Generalisierung

Wie auch die Profilauswertung kann die Generalisierung der Mißbrauchsregeln unabhängig von den restlichen Analyseschritten ausgeführt werden. Nachdem die Regeln ensprechend Kapitel 3 generalisiert wurden, muß entschieden werden, in welchem Umfang diese reduzierte Mißbrauchsregelmenge zum Vergleich mit den zu analysierenden Datensätzen eingesetzt werden soll. Da von einer Mindestkonfidenz ausgegangen werden kann, ist das entscheidende Kriterium die Abdeckung, aber auch die Regelanzahl. Je nach zur Verfügung stehenden Recourcen sollte also die Regelmenge eingeschränkt werden. Als gutes Maß hat sich eine Abdeckung von 80% ergeben, die mit ungefähr 10% der eigentlichen zur Generalisierung verwendeten Mißbrauchsregeln erreicht werden kann (siehe Abschnitt 3.12). Für die Regeln der GL 17 bis GL 4, insgesamt 747 Regeln, konnte eine Abdeckung von 90,91% auf allen vorliegenden Mißbrauchsdaten erreicht werden. Durch die Aussortierung von Regeln mit einer tatsächlich zu kleinen Konfidenz konnte diese Menge auf eine Zahl von nur 510 Regeln dezimiert werden, jedoch sinkt dadurch die Abdeckung auf einen Wert von 83,08%. Nichtdestotrotz konnte eine Trefferquote von 99,64% im ersten Fall und eine Gesamttrefferquote von 99,79% im zweiten Fall auf *allen* vorliegenden Daten erreicht werden. Dabei werden insgesamt 542858 legale und 5850 illegale Daten betrachtet. Die genauen Ergebnisse sind in Abschnitt 3.12.4 nachzulesen.
Desweiteren wurde die Analyse auf Basis der von der Profilanalyse markierten

Daten auf 500 Mißbrauchsdaten und 500 legalen Daten durchgeführt. Ein Training des abschließenden Neurons erübrigt sich an dieser Stelle, da es sich nur um ein einzelnes Datum handelt. Es konnten folgende Werte bei der Klassifikation, die nichts weiter als die in Abschnitt 3.12.4 vorgestellte Regelfilterung darstellt, erzielt werden:

Diagnose	legal	illegal	gesamt	Konfidenz
richtig	100,0% (500)	80,2% (401)	**90,1%**	100,0%
falsch	0,0% (0)	19,8% (99)	9,9%	

Tabelle 6.13: Abdeckungsergebnisse der dezimierten Regeln auf 1000 Datensätzen

Desweiteren wurde eine weitere paritätische Auswertung auf allen Mißbrauchsdaten und somit auch auf 5850 legalen Transaktionsdaten durchgeführt. Diese Ergebnisse sind in Tabelle 6.14 zusammengefaßt.

Diagnose	legal	illegal	gesamt	Konfidenz
richtig	99,98% (5849)	83,08% (4860)	**91,53%**	80,6%
falsch	0,02% (1)	16,92% (990)	8,47%	

Tabelle 6.14: Abdeckungsergebnisse der dezimierten Regeln auf 11700 Datensätzen

Die Trefferquote beträgt in diesem Zusammenhang 90,1% beziehungsweise 91,5%. Anhand der Trefferquoten seitens der legalen Zuweisungen ist zu erkennen, wie empfindlich die Konfidenz auf Fehlalarme reagiert. Schon ein Fehlalarm unter den 500 legalen Datensätzen würde zu einer Konfidenz von nur noch 28,42% führen. Dies liegt an dem Hochrechnungsfaktor für die legalen Daten, der dazu beiträgt, ein realistisches Verhältnis aus legalen und illegalen Daten von 1000:1 zu erzeugen. Nichtdestotrotz liegen die Konfidenzwerte im Bereich des Möglichen, wie die ähnlichen Ergebnisse in Abschnitt 3.12.4 zeigen.

Profilanalyse und Regelfilterung Nun soll die gemeinsame Auswertung der bisher vorgestellten Analyseverfahren vorgestellt werden. Dazu wurden die Ausgaben der einzelnen Methoden zusammengefaßt und gewichtet. Das abschließende lineare Neuron wurde zunächst mit 300 Datensätzen trainiert und konnte dabei eine Trefferquote von 90,7% erreichen. Dieser Wert konnte annähernd in der Verifikationsphase mit 90,1% auf den insgesamt 1000 Daten, jeweils 500 pro Klasse, bestätigt werden. Die genaue Aufschlüsselung der Trefferquoten befindet sich in Tabelle 6.15.
Es läßt sich aufgrund der fehlerfreien Zuordnung der legalen Datensätze daraus eine Konfidenz von 100% errechnen. Wie auch anhand des abschließenden, die beteiligten Zuordnungsentscheidungen wertenden, Gewichtsvektors zu erkennen,

Diagnose	legal	illegal	gesamt	Konfidenz
richtig	100,0% (500)	80,2% (401)	**90,1%**	100,0%
falsch	0,0% (0)	19,8% (99)	9,9%	

Tabelle 6.15: Klassifizierung mit Regelfilterung und den Profilentscheidungen auf 1000 Datensätzen

wirkt sich ausschließlich die Regelfilterung auf das Endergebnis aus. Die Ergebnisse sind identisch mit denen aus Tabelle 6.13. Die Profilauswertung spielt nur eine untergeordnete Rolle, die keinerlei Auswirkung auf das abschließende Ergebnis hat.

Klassifizierung der Analogdaten

Desweiteren wurde eine Klassifizierung der einzelnen Analogwerte, wie in Kapitel 4 beschrieben, durchgeführt. Dazu wurden die sieben verschiedenen Unternetzwerktypen eingesetzt, um jeweils eine binäre Einordnung der zu behandelnden Daten vorzunehmen. Ein abschließendes lineares Neuron gewichtet diesen Binärvektor, so daß es mit Hilfe einer Schwellenwertoperation zu einer Klassenentscheidung kommt. Es wurde zunächst erneut ein Training von 300 Trainingsläufen auf den von der Profilauswertung markierten und verwendeten Datensätzen durchgeführt. Dabei konnte eine Trefferquote von 80,7% erreicht werden. Der anschließende Verifikationsdurchlauf erbrachte die in den Tabellen 6.16 und 6.17 abgebildeten Werte.

Diagnose	legal	illegal	gesamt	Konfidenz
richtig	95,2% (476)	75,4% (377)	**85,3%**	1,547%
falsch	4,8% (24)	24,6% (123)	14,7%	

Tabelle 6.16: Klassifizierung der Analogdaten auf 1000 Datensätzen

Diagnose	legal	illegal	gesamt	Konfidenz
richtig	92,4% (5408)	71,0% (4153)	**81,72%**	0,931%
falsch	7,6% (442)	29,0% (1697)	18,28%	

Tabelle 6.17: Klassifizierung der Analogdaten auf 11700 Datensätzen

Die Trefferquoten betragen, wie in den Tabellen abzulesen, 85,3% auf einer Datenbasis von 1000 Datensätzen und 81,7% auf 11700 Datensätzen.
Die Konfidenz erreicht damit einen Wert von 1,547% im ersteren und 0,931% im letzteren Fall. Im folgenden soll, wie es schon der Fall bei der Profilauswertung war, die Klassifizierung der Analogdaten in Verbindung mit den jeweils einzelnen, vorhergehenden Analysemethoden durchgeführt werden.

Klassifizierung der Analogdaten und Regelfilterung Es werden nun die Resultate der Klassifizierung und zusätzlich der Regelfilterung gemeinsam von dem abschließenden Sigmaneuron gewichtet. Wie im vorangegangenen Fall wird zunächst ein Training auf 300 Datensätzen durchgeführt. Dabei kann eine Trefferquote von 83,3% erreicht werden. Die Verifikationsphase wurde nun wie beim Profil auf 1000 sowie zusätzlich auf 11700 Datensätzen ausgeführt, wobei letzteres dem Maximum an vorliegenden Mißbrauchsdaten entspricht. Durch diese höhere Verifikationsmenge können erneut genauere und allgemein gültigere Werte als auf dem eingeschränkten Datenraum von 1000 Werten erzielt werden. Es konnten dabei folgende Trefferquoten erzielt werden:

Diagnose	legal	illegal	gesamt	Konfidenz
richtig	100,0% (500)	85,6% (428)	**92,8%**	100,0%
falsch	0,0% (0)	14,4% (72)	7,2%	

Tabelle 6.18: Analogdaten und Regelfilterung auf 1000 Datensätzen

Diagnose	legal	illegal	gesamt	Konfidenz
richtig	91,7% (5362)	87,9% (5139)	**89,75%**	1,048%
falsch	8,3% (488)	12,1% (711)	10,25%	

Tabelle 6.19: Analogdaten und Regelfilterung auf 11700 Datensätzen

Das ergibt Gesamttrefferquoten von 92,8% auf der Menge von 1000 Daten und von 89,8% auf der Menge von 11700 Datensätzen. Die neu erzielten Konfidenzwerte betragen für ersteren Fall auf 1000 Daten 100,0% und für letzteren auf 11700 Daten 1,048%. An dieser Stelle wird die extreme Auswirkung der legalen Trefferquote auf die Konfidenz erneut deutlich.

Klassifizierung der Analogdaten mit Profilanalyse Nun kommt es zu der gemeinsamen Auswertung der Klassifizierungsergebnisse der Analogdaten in Verbindung mit den Resultaten bei der Profilanalyse. Erneut wurde wie bisher das zusammenfassende lineare Neuron mit 300 Trainingszyklen auf den von der Profilanalyse bearbeiteten Datensätzen trainiert. Dabei konnte eine Trefferquote von 81,3% erzielt werden. In der anschließenden Verifikationsphase mit 1000 Datensätzen konnte dann eine Trefferquote von 84,1% erreicht werden. Die einzelnen Treffer- beziehungsweise Fehlerquoten sind in Tabelle 6.20 zusammengefaßt. Auf Basis dieser Werte konnte eine Konfidenz von 3,038% berechnet werden.

6.3.4 Gesamtergebnisse

Abschließend soll nun eine Klassifizierung auf Basis *aller* Analysemethoden durchgeführt werden. Dazu wird, wie schon bei den Zwischenergebnissen geschildert,

Diagnose	legal	illegal	gesamt	Konfidenz
richtig	97,6% (488)	75,2% (376)	**86,4%**	3,038%
falsch	2,4% (12)	24,8%(124)	13,6%	

Tabelle 6.20: Klassifizierung der Analogdaten inklusive Profilentscheidungen auf 1000 Datensätzen

verfahren. Zunächst wird das komplette Netz, im wesentlichen die Gewichte des zusammenfassenden linearen Neurons, mit 300 Trainingsdatensätzen für die Klassifizierung vorbereitet. Insgesamt müssen dort also 10 Gewichte antrainiert werden, sieben für die binären Ergebnisse von den einzelnen RBF-Unternetzwerken und 2 weitere für die Ergebnisse der Profilauswertung sowie eines für die Ergebnisse der Regelfilterung. Die Gewichte der einzelnen Eingänge bei einer Klassenschwelle von 0 sind in folgender Tabelle 6.3.4 dargestellt und können als Indiz für die Aussagekraft der einzelnen zu gewichtenden Komponenten angesehen werden. Dies ist möglich, da es sich ausschließlich um binäre Eingabewerte in der Form von -1 für die Einstufung als Mißbrauch und $+1$ für eine Einstufung als legal handelt.

Netztyp	Gewicht
Uhrzeit	0,0948
Beträge	0,2107
Eröffnungsdatum	0,1418
Ablaufdatum	0,0795
Alter	-0,0215
Kreditlimit	-0,1077
Kreditrahmen	0,0847
Profil(analog)	-0,0276
Profil(symbolisch)	0,1688
Regelfilter	0,4363

Tabelle 6.21: Antrainierte Gewichte des abschließenden linearen Neurons

Man erkennt deutlich anhand der Gewichte, welche Netztypen beziehungsweise Analysemethoden entscheidend zu der abschließenden Klassenzuordnung beitragen. Hervorzuheben sind hier das radiale Basisfunktionsnetz zur Betragsanalyse und die Regelfilterung. Das Karteninhaberalter analysierende RBF-Netz sowie das Netz für die Auswertung des Kreditlimits trägt zu einer Klassenentscheidung im negativen Sinne bei. Das heißt, die einzelne Netzentscheidung wird bewußt nicht beachtet und entgegengesetzt verwertet. Dieses Faktum ist damit zu interpretieren, daß zum einen die Unternetzwerke wahrscheinlich noch nicht ausreichend trainiert wurden oder zum anderen eben, daß bei einem nicht Eintreten der von den RBF-Netzen gelernten, legalen Situation ein Mißbrauch wahrscheinlicher ist.

Zusammenfassend wurde nun versucht, die Trefferquoten für die legalen sowie illegalen Daten im ganzen zu optimieren. Insgesamt konnte bei dem paritätischen Training auf 300 Datensätzen, also jeweils 150 aus der legalen bzw. illegalen Klasse, eine Trefferquote von 92,0% erreicht werden.

Die anschließende Verifikation auf 1000 Datensätzen, mit jeweils 500 Daten aus den jeweiligen Datenklassen, führte zu folgenden Ergebnissen:

Diagnose	legal	illegal	gesamt	Konfidenz
richtig	99,4% (497)	84,8% (424)	**92,1%**	12,383%
falsch	0,6% (3)	15,2% (76)	7,9%	

Tabelle 6.22: Trefferquoten und Konfidenz des Komplettmodells auf 1000 Datensätzen

Damit wurde letztendlich eine Trefferquote von 92,1% erreicht. Die daraus resultierende Konfidenz berechnet sich nach Gleichung 3.17 aus Abschnitt 3.5, wobei wie bisher die Fehlerquote der legalen Datenzuordnungen mit dem Faktor 1000 multipliziert werden muß, um ein realistisches Ergebnis in Bezug zum tatsächlichen Datenverhältnis zu erhalten.

6.3.5 Zusammenfassung

Zusammenfassend seien an dieser Stelle noch einmal die Ergebnisse der einzelnen Analysekombinationen aufgeführt, um den Zugewinn der einzelenen Schritte zu verdeutlichen.

Analysemethode	Trefferquote (%)		Konfidenz (%)	
	1000	11700	1000	11700
Profil	68,5	-	0,656	-
Regelfilter	90,1	91,5	100,000	100,000
analoge Daten	85,3	81,7	1,547	0,931
Profil + Regelfilter	90,1	-	100,000	-
analoge Daten + Profil	86,4	-	3,038	-
analoge Daten + Regelfilter	92,8	89,8	100,000	1,048
sämtl. Analysemethoden	92,1	-	12,383	-

Tabelle 6.23: Zusammenfassung der Trefferquoten und Konfidenzwerte der gewichteten Diagnose auf 1000 und 11700 Verifikationsdaten

Im Vergleich sollen auch an dieser Stelle in Tabelle 6.24 die einzelnen Trefferquoten zusammenfassend abgebildet werden, um den Verlauf der einzelnen Erkennungsquoten verfolgen zu können. Hierbei kommt zur Geltung, wie die Trefferquoten der jeweiligen Klassifizierungsschritte im einzelnen und der Zugewinn dieser bei der Kombination der Analysemethoden ausfallen.

Analysemethode	Trefferquoten (1000)		Trefferquoten (11700)	
	legal	illegal	legal	illegal
Profil	93,4	43,6	-	-
Regelfilter	100,0	80,2	99,98	83,08
analoge Daten	95,2	75,4	92,4	71,0
Profil + Regelfilter	100,0	80,2	-	-
analoge Daten + Profil	97,6	75,2	-	-
analoge Daten + Regelfilter	100,0	85,6	91,7	87,9
sämtl. Analysemethoden	99,4	84,8	-	-

Tabelle 6.24: Trefferquoten der einzelnen Schritte auf 1000 und 11700 Verifikationsdaten

6.3.6 Unparitätisches Training

Wie auch schon im ersten Ansatz soll nun erörtert werden, wie sich ein an das Datenverhältnis in der Realität angepaßteres Training auf die anschließende Klassifizierung auswirkt. Dabei kann, wie zuvor erwähnt, nicht das tatsächliche Verhältnis aus 1000:1 legalen und illegalen Datensätzen zum Training verwendet werden, da sonst der Trainingsumfang zu groß geraten würde. Statt dessen soll auch hier wieder ein kleineres Verhältnis gewählt werden.

Es wurde erneut auf 300 Datensätzen trainiert. Dabei wurde ein Trainingsverhältnis von 2:1 benutzt. Das bedeutet genau, auf 2 legale kommt ein illegaler Datensatz bei der Datenrekrutierung zum Training. Bei 300 Datensätzen macht das 200 legale Datensätze und 100 illegale. Nachdem das Netz auf diese Art und Weise trainiert wurde, konnten die Gewichte aus Tabelle 6.25 des abschließenden Neurons gelernt werden.

Netztyp	Gewicht
Uhrzeit	-0,0366
Beträge	0,1557
Eröffnungsdatum	0,1152
Ablaufdatum	-0,0237
Alter	-0,0239
Kreditlimit	-0,0261
Kreditrahmen	0,0713
Profil(analog)	-0,0198
Profil(symbolisch)	0,1287
Regelfilter	0,4511

Tabelle 6.25: Antrainierte Gewichte des abschließenden linearen Neurons durch unparitätisches Training (2:1)

Es konnte eine Gesamttrefferquote beim Training von über 83,0% erreicht werden. Mit der anschließenden Verifikationsphase konnten die Ergebnisse aus Tabelle 6.26 erzielt werden. In der Verifikationsphase wurden jedoch die 1000 Datensätze paritätisch, das heißt gleichverteilt aus der Menge der Datensätze rekrutiert. Ebenso wurde bei einem Training mit dem Trainingsdatenverhältnis von 3:1 verfahren. Die abschließenden Ergebnisse sind ebenfalls in Tabelle 6.26 abgebildet.

Trainingsverhältnis	legal	illegal	gesamt	Konfidenz
2:1	100,0% (500)	81.2% (406)	**90,6%**	100,0%
3:1	100,0% (500)	79.6% (398)	**89,8%**	100,0%

Tabelle 6.26: Trefferquoten und Konfidenz des Komplettmodells auf 1000 Datensätzen, mit unterschiedlichen Trainingsverhältnissen

6.3.7 Auswertung

Zunächst soll auf die einzelnen Ergebnisse der Modellherleitung eingegangen werden. Es ist anhand der Werte in Tabelle 6.23 zu erkennen, daß die Trefferquote durch die Kombination der verschiedenen Analysemethoden erhöht werden können. Auffällig ist, daß die Abarbeitung aller Analysemethoden *nicht* das Maximum der Gesamttrefferquoten erreicht. Dieser Trend ist ebenso bei den einzelnen Trefferquoten zu erkennen. Im Zusammenhang mit den legalen Daten führt eine Analyse auf Basis des Regelfilters jedoch immer zu maximalen Ergebnissen. Erst die abschließende Verrechnung der einzelnen Unterergebnisse kann dieses Maximum nicht erreichen, wie auch an den Konfidenzwerten zu ersehen ist. Es ist zu vermuten, daß durch die hohe Anzahl an konkurrierenden Klassifizierungsergebnissen, es zu einem Übersprechen der guten Regelfilterungsentscheidung kommt, und somit Fehlalarme erzeugt werden.
Die Trefferquoten auf einer größeren Regelmenge von 11700 Datensätzen fallen generell etwas schlechter aus als die Ergebnisse auf den nur 1000 Datensätzen. Dennoch ist auch hier der Trend zur Verbesserung durch die Kombination aus Regelfilter und Analogdatenauswertung bezüglich der Trefferquoten zu verzeichnen. Es ist denkbar, daß die Ergebnisse aufgrund eines nicht ausreichenden Trainings im Verhältnis zu dem zu klassifizierenden Datenumfang schlechter ausfallen.
Die Trefferquoten bezüglich des unparitätischen Trainings bestätigen die Ergebnisse aus Abschnitt 4.5.3. Mit einer Zunahme an legalen Trainingsdaten steigt die Anzahl der legalen Treffer und damit die Konfidenz, aber die Mißbrauchstrefferquoten fallen langsam ab.

Allgemein zeigen die Ergebnisse eine Diskrepanz zwischen den Werten der Trefferquote und den Konfidenzwerten auf. Mit zunehmender Trefferquote seitens der

Mißbräuche kommt es zu einer Stagnation oder sogar Steigerung der Fehlerquote bei den legalen Zuordnungen. Letzteres hat jedoch wiederum zur Folge, daß die Konfidenz stark abfällt. Man spricht in diesem Zusammenhang von einem sogenannten „Trade-Off“. Abbildung 6.7 auf Seite 145 versucht, den Zusammenhang anhand einer einfachen Klassenentscheidung zu skizzieren.
Im Zusammenhang mit der Mißbrauchsanalyse handelt es sich bei den Kriterien der Trefferquote und der Konfidenz um zwei verschiedene Blickwinkel bei der Klassifizierung, die im Zuge einer Optimierung zu verschiedenen, entgegengesetzten Resultaten führen. So handelt es sich bei der Konfidenz um ein Vertrauensmaß bezüglich der Mißbrauchszuweisung. Im Zuge der Mißbrauchserkennung kann eine hohe Konfidenz als eine Verminderung der Falschzuweisungen im Falle der legalen Transaktionen, also von Fehlalarmen, gedeutet werden. Das heißt, für den Kunden kommt es im Zweifelsfall zu einer Abnahme der irrtümlicherweise abgewiesenen Autorisierungsanfragen. Das bedeutet, es findet eine Stärkung des Vertrauens in die Mißbrauchsentscheidung statt, wobei allerdings die tatsächliche Mißbrauchserkennung unter Umständen abnimmt.
Anders verhält es sich bei der Trefferquote bezüglich des Mißbrauchs. Mit einer höheren Trefferquote auf seiten des Mißbrauchs ist, wie in den Werten zu erkennen, eventuell nur eine mäßige Erhöhung der Trefferquoten auf legaler Seite zu verzeichnen. Hinzu kommt, daß Falschzuweisungen bezüglich der legalen Daten wahrscheinlicher werden. Da jedoch ein derart ungleiches Verhältnis aus illegalen und legalen Daten (1:1000) vorliegt, können die Konfidenzwerte keinen großen Zugewinn erfahren; im Zweifelsfall sogar einen stark ausgeprägten Verlust. Für den Praxisbezug bedeutet das, daß mit einer Trefferquotenerhöhung seitens der Mißbräuche zwar vermehrt Mißbrauch abgefangen werden kann, jedoch zu Lasten der Verläßlichkeit bei der eigentlichen Mißbrauchsentscheidung – der Entscheidung für Mißbrauch.
Es ist nun die Frage nach der Bewertung der einzelnen Einbußen gegeben: Entweder kommt es zu einem finanziellen Schaden durch irrtümliche Kreditverweigerungen, beziehungsweise durch die fälschlicherweise erfolgte Verweigerung der Kreditanforderung – schlimmstenfalls sogar zu einem Verlust des Kunden – oder durch den gegebenen reellen Mißbrauch. Da bei einer fälschlicherweise abgelehnten Autorisierungsanfrage ebenfalls Unkosten für das Kreditinstitut entstehen, wird vorrangig zugunsten des Service am Kunden entschieden, auch mit dem Risiko, mögliche Mißbräuche nicht erkennen und somit abwehren zu können.

Wie auch schon in Abschnitt 4.5.3 geschildert, kann auch in diesem Fall die Trefferquote in Bezug auf die legalen Datensätze bis auf 100% angehoben werden, wenn das Trainingsverhältnis zugunsten dieser legalen Daten zunimmt. Interessant in diesem Zusammenhang ist, daß mit der Zunahme der legalen Trainingsdaten, die Gewichtung des Regelfilters zunimmt und parallel dazu die Gewichtungen der übrigen Analysemethoden abnimmt (Tabelle 6.23). Dieses Faktum führt zu dem Schluß, daß die Trefferquote des Regelfilters bezüglich der lega-

len Datensätze nahezu optimal ausfällt. Dieses Ergebnis konnte ebenfalls schon in Abschnitt 3.12.4 erzielt werden. Auch die Konfidenzwerte zeigen offensichtliche Parallelen zu den Ergebnissen aus Abschnitt 3.12.4 auf. Aufgrund dieser Dominanz der regelspezifischen Klassenentscheidung bezüglich der binären Ausgabe der einzelnen Analyseschritte kann man fast von von einer „Konjunktion“ der Ergebnisse aus Regelfilterung und den übrigen Analysemethoden sprechen. Kommt es zum Beispiel zu einer Mißbrauchsentscheidung des Regelfilters, müßte die Entscheidung der übrigen Analysemethoden geschlossen dagegen entscheiden, um eine derartige Bewertung zu falsifizieren. Auch für den umgekehrten Fall gilt dies. Für die komplette Analyse bedeutet das, daß mit dem Ziel der Optimierung der Konfidenz von Anfang an vermehrt Gewicht auf die Entscheidung durch die Filterung der Daten mit den generalisierten Regeln gelegt werden muß, und die übrigen Entscheidungsfindungen unter Umständen nur korrigierend im Falle eines möglichen Mißbrauchs einschreiten.

Unter Kenntnis dieser Voraussetzung kann das vorgestellte Modell jedoch nicht als optimal bezeichnet werden, wenn in Folge der Hinzunahme neuronaler Netze, die Konfidenz abnimmt, weil die Klassenentscheidung des Regelfilters bezüglich der legalen Datensätze durch die RBF-Netze im Extremfall negativ beeinflußt wird. Hinzu kommt, daß es fast nicht möglich ist, ein neuronales Netz mit einer Trefferquote besser als 99,9%, wie sie allein durch das Datenverhältnis von 1:1000 gegeben ist, zu entwickeln, wenn die zu analysierenden Klassen sich überschneiden. Aus diesem Grund wird in jedem Fall durch eine ungünstige Kombination aus einzelnen Entscheidungen das sehr gute Klassifizierungspotential der Regelfilterung durch die generalisierten Regeln verschlechtert und im ungünstigsten Fall sogar zunichte macht. Als konsequente Schlußfolgerung kann nur gelten, daß die Regelfilterung nicht „konkurrierend“ zu den neuronalen Netzen, den Analogdaten auswertenden Analysemodellen, eingesetzt wird, sondern explizit durch diese in Zweifelsfällen nur ergänzend, beziehungsweise unterstützend eingesetzt wird. Dies führt zu einem zweiten regelbasierten Ansatz, der nun folgend in Abschnitt 6.4 vorgestellt wird.

6.4 Das hierarchische Modell

6.4.1 Motivation und Aufbau

Mit diesem dritten Ansatz zur Mißbrauchsprävention soll versucht werden, die Nachteile der vorangehenden Modelle zu beheben. Wie im regelspezifischen Analysemodell aus Abschnitt 6.2 wird erneut eine hierarchische Verarbeitungsstruktur verwendet, die im Unterschied dazu wie im Modell aus Abschnitt 6.2 auf einer größeren Menge an generalisierten Regeln arbeitet. Desweiteren kann durch eine Staffelung der einzelnen Analyseschritte der Aufwand besonders mit Hinblick auf die ergänzende Analogdatenauswertung erheblich reduziert werden. Weiter kann auf diese Weise der negative Einfluß der Analogentscheidungen, wie er im parallelen Modell (Abschnitt 6.3) vorkommt, vermieden werden. Trotzdem ist der Aufbau dieses Modells prinzipiell auf Basis der bisher vorgestellten Analysemodule einschließlich der Profilauswertung realisierbar. Abbildung 6.4 zeigt die Abfolge der Auswertungen, bei der die allgemeine Regelfilterung im Vordergrund steht.

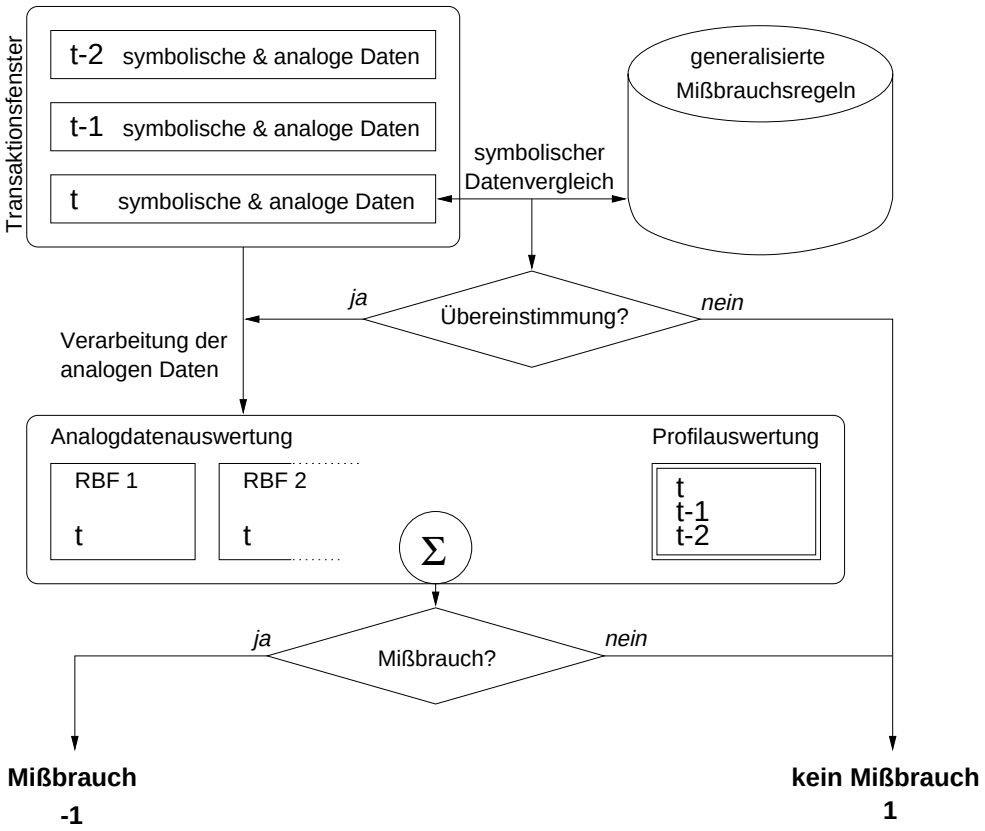


Abbildung 6.4: Flußdiagramm des 2. Ansatzes

In der Abbildung sind die zwei gestaffelten Entscheidungsknoten zu erkennen. So findet im oberen Knoten, der mit `Übereinstimmung?` betitelt ist, die Regelfilterung auf Basis der generalisierten Regeln statt. Es handelt sich dabei um eine Einordnung der Datensätze mit der durch die abdeckenden Regeln gegebenen Konfidenz. Kommt es bei diesem Vergleich zu einer Gleichheit zwischen dem aktuellen Datensatz `t` (in Abbildung 6.4), so liegt ein potentieller Mißbrauch vor, und es schließt sich die Verarbeitung der Analogdaten an. Diese wird entsprechend der vorhergehenden Abschnitte mit Hilfe der RBF-Netze durchgeführt. Mit ein Bestandteil dieser Auswertung ist die Profilauswertung, bei der entsprechend der betrachteten Fenstergröße (3) die Transaktionshistorie durch die Datensätze `t-1` und `t-2` mit in die Klassifizierung einbezogen wird. Die abschließende Gewichtung und Klassifizierung der einzelnen Netzentscheidungen entscheidet mit dem zweiten Knoten `Mißbrauch?` endgültig über die Mißbrauchszuweisung.
Es wird also im Falle eines potentiellen Mißbrauchs der aktuelle Datensatz mit Hilfe der Analogdaten genauer und eingehender untersucht, bevor es zu einer abschließenden Beurteilung der Transaktion kommt. Zum Training dieser Diagnose auf Basis der Analogdaten können prinzipiell unabhängig trainierte Netze, zum Beispiel die aus den Kapiteln 4 und 5, hinzugezogen werden.
Durch dieses Vorgehen kann die Konfidenz im wesentlichen durch Reduzierung der fälschlicherweise durch die Regelfilterung als Mißbrauch bezeichneten, legalen Transaktionen erhöht werden, wie bildlich auch anhand Abbildung 6.7 auf Seite 145 nachzuvollziehen ist. Die Regelfilterung wird also *nicht* wie im parallelen Ansatz aus Abschnitt 6.3 durch eine Gewichtung der einzelnen Methoden in die Klassenentscheidung konkurrierend mit einbezogen. Außerdem wird durch die eventuell genauere Diagnose der potentiellen Mißbräuche die Fehlalarmrate weiter gedrosselt.

Zunächst sollen die einzelnen Ergebnisse, die unter verschiedenen Trainings- und Arbeitsmethoden erzielt werden können, in Abschnitt 6.4.2 vorgestellt werden. Anschließend findet eine Auswertung und Analyse dieser Ergebnisse statt (Abschnitt 6.4.3), bei der auch Vergleiche mit den bisherigen Modellansätzen diskutiert werden.

6.4.2 Ergebnisse der hierarchischen Auswertung

Ähnlich dem Vorgehen in Abschnitt 6.3 soll auch hier eine schrittweise Erweiterung durch die verschiedenen Analysemethoden Aufschluß über die Zugewinne bei den Ergebnissen geben. Aus diesem Grund wird die Profilanalyse zunächst nicht betrachtet, sondern erst zum Ende diese Abschnitts in die weitere Auswertung mit einbezogen.
Die Klassifizierung bezüglich des Regelfilters wird zunächst auf Basis der Regeln des GL 17 bis zum GL 4 durchgeführt. Anschließend wird ebenso wie in Abschnitt 6.3 die Regelfilterung auf der reduzierten Regelmenge durchgeführt, bei der Re-

geln mit einer tatsächlich zu geringen Konfidenz beziehungsweise durch andere Regeln abgedeckte entfernt wurden. Die Ergebnisse, die allein bei der Regelfilterung erzielt werden können, entsprechen denen in Abschnitt 3.12 und 6.3.
Kommt es nun zu einer Übereinstimmung mit *mindestens einer* der verallgemeinerten Regeln, so wird nur dann eine zusätzliche Klassifizierung mit Hilfe des gesamten RBF-Netzapparats, bestehend aus den in Abschnitt 4.4 beschriebenen Unternetzwerken, durchgeführt. Zum Training des Gesamtnetzes wurde zu Vergleichszwecken entsprechend Abschnitt 6.3 eine Datenbasis von 300 Datensätzen, also 150 legalen sowie illegalen Daten, verwendet. Dabei wurden sowohl die RBF-Parameter gelernt als auch die Gewichte des abschließenden linearen Neurons mit einer Schwellenwertoperation. Die Ergebnisse dieses Trainings können ebenfalls in Abschnitt 6.3 nachgeschlagen werden, da die dort trainierten Netze übernommen wurden.

Klassifizierung ohne Profilanalyse

Nun wird auf der maximalen Anzahl von 11700 Datensätzen, die zur paritätisch ausgewerteten Verifikation eingesetzt werden können[2], eine Analyse durchgeführt. Außerdem wird mit Hinblick auf die zusätzliche Profilauswertung die Analyse auf einer eingeschränkten Transaktionsmenge von 1000 Datensätzen durchgeführt, um später vergleichbare Ergebnisse zur Verfügung zu haben. Dabei ist sichergestellt, daß sämtliche Mißbrauchstransaktionen, und ein „grober“ Querschnitt der legalen Transaktionsdaten betrachtet wird. Letzteres geschieht durch eine mit jedem Schritt zufällig gewählte Schrittgröße, mit der sequentiell beliebig oft über die Menge der bisher unbenutzten legalen Daten gegangen wird. Die Ergebnisse der Auswertung auf Basis aller Regeln der GL 17 - 4 und der reduzierten Regelmenge sind in den Tabellen 6.27 und 6.28 aufgeführt:

Datensätze	Diagnose	legal	illegal	gesamt	Konfidenz
1000	richtig	99,8%(499)	80,2%(401)	90,0%	28,622%
	falsch	0,2%(1)	19,8%(99)	10,0%	
11700	richtig	99,92%(5845)	82,14%(4805)	91,03%	49,006%
	falsch	0,08%(5)	17,86%(1045)	8,97%	

Tabelle 6.27: Regelfilter auf Basis von 747 Regeln , GL 17 - GL 4 (1000/11700 Datensätze untersucht)

Man erkennt deutlich an den Konfidenzwerten, daß die Einstufung in die legalen Daten sehr sicher und zuverlässig arbeitet. Dennoch gilt: Wenn nur eine kleine Anzahl an Fehlalarmen auftaucht, fällt die Konfidenz rapide ab, wie bei der Regelfilterung auf Basis der reduzierten (510) Regeln zu erkennen ist. Hier ist

[2]Es existieren nur 5850 illegale Transaktionsdatensätze

Datensätze	Diagnose	legal	illegal	gesamt	Konfidenz
1000	richtig	100,0%(500)	69,0%(345)	84,5%	100,0%
	falsch	0,0%(0)	31,0%(155)	15,5%	
11252	richtig	99,98%(5849)	75,25%(4402)	87,62%	81,488%
	falsch	0,02%(1)	24,75%(1448)	12,38%	

Tabelle 6.28: Hierachisches Analysemodell auf Basis der 510 Regeln, der reduzierten Regelmenge der GL 17 - GL 4 (1000/11700 Datensätze untersucht)

eine falsch zugeordnete Transaktion für einen Konfidenzverlust von knapp 20% verantwortlich. Im Falle der nur 1000 Datensätze wirkt sich das Auftreten eines Fehlalarms jedoch gravierender auf den Konfidenzwert aus, der an dieser Stelle sogar um 70% abfällt. Auch wenn Werte im Bereich von 100% im Falle der Konfidenz sicherlich nicht sehr aussagekräftig sind, können diese in Anlehnung an Abschnitt 3.12.4 bekräftigt werden. Dieser dramatische Trade-Off zwischen Trefferquote und Konfidenz hängt mit dem Hochrechnungsfaktor von 1000 zusammen, der benutzt wird, um den Umfang der legalen Daten auf das reale Verhältnis 1:1000 aus illegalen und legalen Daten wieder herzustellen. Die Konfidenz verhält sich dadurch entsprechend der Funktion $1/x$.

Die Gesamttrefferquoten liegen jedoch in etwa auf dem gleichen Niveau wie jene des ersten Ansatzes in Abschnitt 6.3. Doch trotz der ausreichend guten Konfidenzwerte, ist die Trefferquote um rund 70% seitens der Mißbräuche noch verbesserungsfähig, wie der Vergleich zu den Ergebnissen in Abschnitt 6.3 zeigt: Dort können Mißbrauchstrefferquoten von 80% bei der kombinierten Auswertung durch Regelfilter und analogen Daten erreicht werden. Die erreichten Ergebnisse sind eher vergleichbar mit den Ergebnissen der einfachen Regelfilterung, wie sie ebenfalls im konkurrierenden Modell vorgestellt wurden. Daraus läßt sich schließen, daß ein Hauptteil der Fehlentscheidungen bezüglich der Mißbrauchstransaktionen im ersten Entscheidungsknoten stattfinden. Dies bekräftigen die neben der Klassifizierung protokollierten Fehlentscheidungen, die in Tabelle 6.29 abgebildet sind.

Regelbasis	Datenbasis	Fehler durch Regelfilterung (Knoten 1)	Fehler durch Klassifizierung (Knoten 2)
GL 17 - 4	1000	47 (4,7%)	53 (5,3%)
	11700	532 (4,6%)	518 (4,4%)
reduzierte GL 17 - 4	1000	99 (9,9%)	53 (5,3%)
	11700	990 (8,5%)	459 (3,9%)

Tabelle 6.29: Verteilungen der Fehlentscheidungen

Bei den „Fehlern durch Regelfilterung“ handelt es sich um bei der Regelfilte-

rung als irrtümlich legal angesehene Mißbrauchsdatensätze, die keine Übereinstimmung mit den generalisierten Mißbrauchsregeln aufweisen oder um legale Datensätze, die von mindestens einer der Regeln abgedeckt wurden. Erstere werden nicht weiter bearbeitet, und führen somit zu Falschzuweisungen.
Die „Fehler durch Klassifizierung“ entstehen dadurch, daß zunächst als Mißbrauch ausgezeichnete Datensätze auf Basis der Analogdatenklassifizierung abschließend irrtümlich doch als legal ausgezeichnet werden.
Man erkennt also auch in Tabelle 6.29, daß der Hauptanteil für die fehlerhafte Zuweisung bei der Regelfilterung, besonders im Falle der reduzierten Regeln, liegt, da hier viele Datensätze irrtümlich als legal eingestuft werden. Dies ist ein Problem der Abdeckung der Mißbrauchsregeln. Um hier die Regelfilterung seitens der illegalen Trefferquoten sicherer zugestalten, und trotzdem hohe Konfidenzwerte zu erzielen, ist es denkbar, die Regelmenge zu vergrößern oder die Regeln weiter zu verallgemeinern. Ersteres ist in folgendem Abschnitt durch Hinzunahme der Regeln zweier weiterer GL durchgeführt worden.

Vergrößerung der Regelmenge

Zusätzlich zu den bisher verwendeten Regeln bei der Regelfilterung sollen nun die Regeln des GL 3 und GL 2 zu der Regelmenge hinzugenommen werden. Damit wird der Regelumfang auf eine Zahl von 837 Regeln erweitert. Die Analyse erfolgt nun erneut auf 11700 Datensätzen sowie außerdem wieder auf 1000 ausgesuchten Transaktionsdatensätzen von Konten mit ausreichend Transaktionen. Die Ergebnisse sind in Tabelle 6.30 aufgeführt.

Datensätze	Diagnose	legal	illegal	gesamt	Konfidenz
1000	richtig	99,8%(499)	81,8%(409)	90,8%	29,028%
	falsch	0,2%(1)	18,2%(91)	9,2%	
11252	richtig	99,95%(5847)	84,3%(4930)	92,11%	62,169%
	falsch	0,05%(3)	15,7%(920)	7,89%	

Tabelle 6.30: Regelfilter auf Basis von 837 Regeln , GL 17 - GL 2 (500/11700 Datensätze untersucht)

Man erkennt an den Werten, daß die Trefferquoten nur leicht zugenommen haben. Stark wirkt sich die höhere Trefferquote seitens der legalen Datensätze auf die Konfidenz aus, die aufgrund 2 Fehlalarmen weniger einen Wert von 82,19% annimmt statt zuvor 59,73%. Auch die Trefferquote bezüglich der illegalen Klassenzuweisungen kann leicht angehoben werden. Anhand von Tabelle 6.31 ist jedoch eine leichte Verbesserung bezüglich der Fehlentscheidungen der Regelfilterung im ersten Knoten zu erkennen. Dies Tatsache ist die logische Konsequenz der Vergrößerung der Mißbrauchsregelmenge.
Aus diesem Grund folgt nun in der nächsten Sektion die Auswertung inklusive

Regelbasis	Datenbasis	Fehler durch Regelfilterung (Knoten 1)	Fehler durch Klassifizierung (Knoten 2)
GL 17 - 2	1000	25 (2,5%)	67 (6,7%)
	11700	394 (3,4%)	529 (4,5%)

Tabelle 6.31: Verteilungen der Fehlentscheidungen

der Profilauswertung.

Klassifizierung inklusive der Profilauswertung

Zusätzlich zu der Auswertung der Analogdaten, soll jetzt die Profilauswertung mit in die endgültige Mißbrauchsentscheidung des 2. Knotens hinzugezogen werden. Aufgrund der eingeschränkten Konten (besonders im Fall der gesperrten Konten) mit einer ausreichenden Anzahl an Transaktionen für die benötigte Transaktionshistorie, soll nun ein Verifikationslauf auf der Menge von 1000 Transaktionen, also jeweils 500 legalen sowie illegalen, ausgeführt werden. Die resultierenden Ergebnisse sind zusammengefaßt in Tabelle 6.32 dargestellt. Um den Vergleich zu den vorhergehenden Klassifizierungsläufen herzustellen, sind auch die Fehlentscheidungen der einzelnen Entscheidungsknoten protokolliert und in Tabelle 6.33 zusammengetragen.

Datensätze	Diagnose	legal	illegal	gesamt	Konfidenz
GL 17 - 4	richtig	100,0%(500)	87,2%(436)	93,6%	100,0%
	falsch	0,0%(0)	12,8%(64)	6,4%	
reduziert GL 17 - 4	richtig	100,0%(500)	84,6%(397)	89,7%	100,0%
	falsch	0,0%(0)	15,4%(103)	10,3%	
GL 17 - 2	richtig	100,0%(500)	95,0%(475)	97,5%	100,0%
	falsch	0,0%(0)	5,0%(25)	2,5%	

Tabelle 6.32: Komplette Auswertung auf 1000 Datensätzen untersucht

Zunächst sind die bei dieser nun vollständigen Analyse der Transaktionsdaten die optimalen Konfidenzwerte sowie die überragenden Trefferquoten zu erwähnen. Die Einstufung bezüglich der legalen Daten konnte auf eine 100% Einordnungssicherheit gebracht werden. Zwar wurde in diesem Test nur eine kleine Menge von legalen Daten betrachtet (500 Datensätze), jedoch haben die vorhergehenden Testläufe gezeigt, daß bezüglich der Trefferquoten nur minimale Unterschiede zwischen den Ergebnissen auf einer großen und einer eingeschränkten Regelmenge bestehen. Auch die Trefferquoten seitens der Einstufung bezüglich der illegalen Transaktionsdaten konnte auf sehr gute Werte angehoben werden. Durchgehend konnte eine Verbesserung dieser Trefferquote erzielt werden, allein durch die Hinzunahme der Profilauswertung in die unterstützende Analyse der analogen Daten.

Erfolgsquoten von weit über 80% ermöglichen es eine gute Mißbrauchserkennung zu gewährleisten.
Die Konfidenzwerte erreichen damit trotzdem den optimalen Wert von 100%. Wie die bisherigen Tests gezeigt haben, führen jedoch kleine Schwankungen seitens der Trefferquote bei den legalen Datensätzen zu ausgeprägten Schwankungen bezüglich der Konfidenz. So kann unter Umständen davon ausgegangen werden, daß bei einer größeren Regelmenge die Konfidenzwerte abfallen auf Werte wie sie in den vorangegangenen Testläufen auf größeren Datenmengen, zum Beispiel den Ergebnissen aus den Tabellen 6.27, 6.28 und 6.30, erzielt werden konnten.

Regelbasis	Fehler durch Regelfilterung (Knoten 1)	Fehler durch Klassifizierung (Knoten 2)
GL 17 - 4	47 (4,7%)	17 (1,7%)
GL 17 - 4 (reduziert)	99 (9,9%)	4 (0,4%)
GL 17 - 2	25 (2,5%)	0 (0,0%)

Tabelle 6.33: Verteilungen der Fehlentscheidungen: Auswertung inklusive Profilanalyse auf 1000 Daten

Wie man an den Ergebnissen, die inklusive der Profilauswertung erzielt wurden, erkennt, kann auf Seiten der Klassifizierung, die erst bei einer Mißbrauchseinstufung der Transaktionen im Falle der symbolischen Daten eintritt, eine verminderte Fehlerhäufigkeit festgestellt werden. Die Werte konnten durchgehend um einen wesentlichen Anteil verringert werden, die Klassifizierung konnte bis zur fehlerfreien Einstufung hin optimiert werden (erweiterter Regelsatz der GL 17 - 2). Logischerweise ist seitens der symbolischen Einstufung durch den Regelfilter bei dieser erweiterten Analyse keine Verbesserung festzustellen, dennoch kommt es zu einer bestätigenden Übereinstimmung bezüglich der erzielten Werte mit zuvor ermittelten Fehlerquoten zum Beispiel aus Abschnitt 3.12.4. Mit Hilfe der Analogdaten konnte also der Fehleranteil, der fälschlicherweise zu Mißbrauch geführt hätte nahezu komplett beseitigt werden.

6.4.3 Zusammenfassung und Auswertung

Das vorgestellte Modell vereint am besten die hohe Zuordnungssicherheit auf Basis der generalisierten Mißbrauchsregeln sowie die Klassifizierung der Analogdaten mit Hilfe der neuronalen Netze. Durch die hohen Konfidenzwerte, die auf einer hohen Trefferquote bezüglich der legalen Datensätze basieren, ist die Möglichkeit gegeben, schnell und effizient eine sichere „Vorsortierung“ der Transaktionsdatensätze vorzunehmen. Damit ist das Risiko eines Fehlalarms auf ein Minimum reduziert. Es konnten insgesamt die besten Resultate bezüglich der Trefferquote der legalen Transaktionsdaten erzielt werden, vergleicht man die drei vorgestell-

ten Ansätze.
Weiter besteht die verwendete Regelmenge, wie in Kapitel 3 gezeigt, aus nur einem zehntel der eigentlichen Mißbrauchsdaten, etwa 600 Regeln. Auf diese Weise kann eine Aussortierung der potentiellen Mißbrauchsregeln sehr schnell durchgeführt werden. Trotzdem können entsprechend der Erkenntnisse in Kapitel 3 bis über 80% der bekannten Mißbräuche auf diese Weise abgedeckt und erkannt werden.

Kommt es zu solch einem möglichen Mißbrauch, so wird genauer auf die Eigenschaften der Transaktion mit Hilfe der neuronalen Netze eingegangen. Hier entscheidet sich dann erneut, diesmal aufgrund der analogen Daten, ob eine Autorisierung stattfindet oder nicht. Durch diesen Schritt wird also die Mißbrauchstrefferquote optimiert, indem die bestmöglichen Resultate aus den bei der Regelfilterung selektierten Datensätze herausgeholt werden. Dies zeigt sich an dem leichten Anstieg um knapp ein Zehntel der Mißbrauchstrefferquoten bei Hinzunahme der Profilauswertung in Tabelle 6.32. Auch bezüglich der Fehlentscheidungen in dieser Entscheidungsebene ist durch die zusätzliche Profilauswertung eine Abnahme um nahezu die Hälfte zu verzeichnen. Im Falle der erweiterten Regelmenge kann die abschließende Klassifizierung des 2. Knotens dann sogar fehlerfrei arbeiten – der Fehler liegt dann ausschließlich bei der durch die Regelfilterung entschiedene Zuordnung. Dies läßt sich anhand der falsch zugewiesenen illegalen Datensätze in Tabelle 6.32 und der Anzahl der Fehlentscheidungen bei der Regelfilterung in Tabelle 6.33 erkennen. Dennoch können sowohl bei der Gesamttrefferquote als auch bei der Konfidenz Zuwachsraten aufgrund der hinzugenommenen Profilauswertung erreicht werden.

Interessant ist in diesem Zusammenhang, daß sobald die Mißbrauchsregelmenge eingeschränkt wird, das heißt Regeln aussortiert werden, die Mißbrauchstrefferquote ansteigt. Durch die großzügigere Einstufung in legale Transaktionen profitiert die Trefferquote bezüglich der legalen Transaktionen und damit in entscheidendem Ausmaße die Konfidenz. Diese wird auch hier mit dem für die legalen Transaktionen benötigten Faktor 1000 berechnet, um das reale Verhältnis aus legalen und illegalen Daten zu wahren. Auf der anderen Seite werden auf diese Weise ohne eine genauere Prüfung durch die analogdatenverarbeitende Schicht, eigentliche Mißbrauchstransaktionen der legalen Klasse zugeordnet, was mit einem Verlust der Trefferquote seitens der Mißbräuche verbunden ist. Also auch hier kann die primäre Intention bei der Analyse durch verändern der Regelmenge reguliert werden. Durch eine umfangreichere Mißbrauchsregelmenge kann vermehrt Mißbrauch aufgedeckt werden, wohingegen durch eine eingeschränkte Mißbrauchsregelmenge die Konfidenz gesteigert, daß heißt die Fehlalarme reduziert werden.

6.5 Auswertung und Diskussion

Wie die Ergebnisse zeigen, können die besten Ergebnisse mit Hilfe des zuletzt diskutierten Ansatzes aus dem Abschnitt 6.4 erzielt werden. Mit diesem Ansatz wird explizit die Regelfilterung separat von der Analogdatenanalyse durchgeführt, wie es auch im regelbasierten Ansatz aus Abschnitt 6.2 vorgesehen ist. Darauf aufbauend wird im Gegensatz zum konkurrierenden Ansatz aus Sektion 6.3 im Zweifelsfalle eines potentiellen Mißbrauchs dann die Profilanalyse und die Analyse der Analogdaten durchgeführt. Die Möglichkeit, daß es trotz der Gewichtung der einzelnen Analysemethoden im abschließenden Neuron zu einem übersprechen der richtigen Entscheidung durch falsche Teilentscheidungen, wie im Falle des zweiten, konkurrierenden Modells, kommt, ist damit nur im Mißbrauchsfall gegeben. Auch im ersten, regelbasierten Modell aus Abschnitt 6.2 ist eine revidierende Klassenentscheidung durch die Profil- und Analogdatenanalyse möglich. Das Problem des Übersprechens beim konkurrierenden Modell wird zum Beispiel deutlich bei der Tatsache, daß die Trefferquote des Regelfilters kombiniert mit der Analogdatenauswertung ohne Profilanalyse im Falle des paritätischen Trainings besser ist, als die *aller* Teilanalysen gemeinsam im Klassifikationsmodell (Tabelle 6.23). Im Gegensatz dazu ist die im letzten Modell konstruierte Analysehierarchie auf die Optimierung der Konfidenz ausgelegt. Die Abbildungen 6.5 und 6.6 machen dies nocheinmal deutlich.

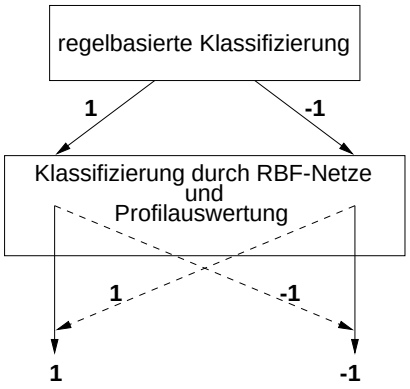


Abbildung 6.5: Modell mit Möglichkeit zur kompletten Korrektur der regelbasierten Entscheidung

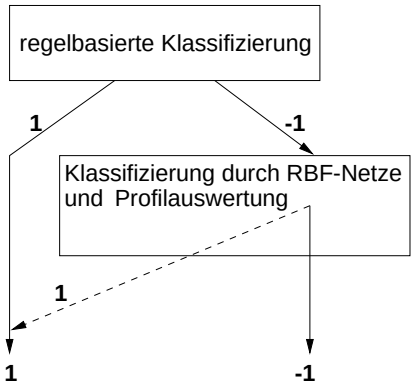


Abbildung 6.6: Modell mit der Möglichkeit, nur im Falle einer Mißbrauchseinstufung durch die Regelfilterung eine korrigierende Entscheidung zu treffen

6.5.1 Konfidenzauswertung

Ein wesentliches Kriterium bei der Auswertung der Klassifizierungsergebnisse ist die Konfidenz bezüglich des Mißbrauchs, die als Maß für die Zuverlässigkeit

der Mißbrauchsentscheidung gilt. Der unter anderem in Abschnitt 6.3.7 zitierte Trade-Off von Trefferquote und Konfidenz wird anhand folgender Skizze deutlich:

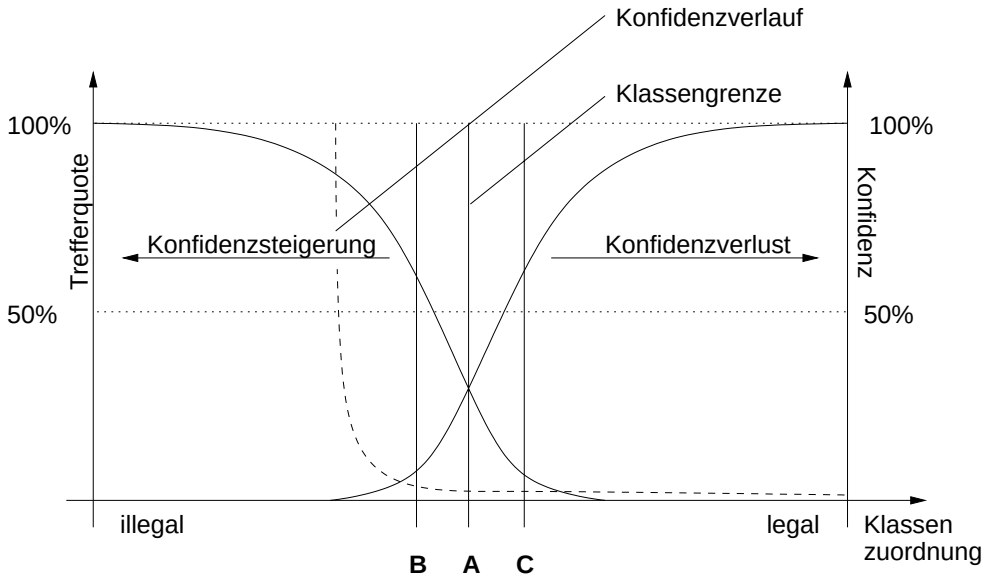


Abbildung 6.7: Trade-Off von Trefferquote und Konfidenz

Im Klassifikationsansatz aus Abschnitt 6.3 kann bei einem paritätischen Training nur ein sehr kleiner Konfidenzwert von maximal 12% erreicht werden, indem alle zur Verfügung stehenden Analysemethoden zusammen zur Mißbrauchszuordnung eingesetzt werden. Im Vergleich dazu wurden mit Hilfe der regelbasierten Ansätze Konfidenzwerte von maximal 100% erreicht. Dieser Wert von 100% wird erreicht, durch die Tatsache, daß gewöhnlich *kein* Fehlalarm registriert, kein legaler Datensatz fälschlicherweise als Mißbrauch zugeordnet wurde. Für Abbildung 6.7 bedeutet dies, daß die Klassengrenze ausreichend weit in Richtung `B` verschoben wurde, so daß sämtliche Entscheidungen für eine legale Transaktion ohne Einschränkung gefällt werden. Trotzdem ist dieser Wert kritisch zu betrachten, da er auf eingeschränkten Datenmengen in den entsprechenden Verifikationsphasen erzielt wurde. Es ist leicht möglich, daß es sich trotz der Bemühung, eine möglichst repräsentative Datenauswahl zu treffen – besonders im Falle der legalen Daten –, um einen zu kleinen Datensample handelt, auf dem die Simulation durchgeführt wurde. Es konnten ähnliche Erfahrungen im Zuge der Generalisierung gemacht werden, wo ebenfalls die Teilmengen mit einer Größe von 30000 Datensätzen nicht ausreichend repräsentativ waren.
Nichtdestotrotz können jedoch Rückschlüsse bezüglich der Konfidenz bei den regelbasierten Ansätzen auf die in Kapitel 3 Abschnitt 3.12.4 auf Seite 60 ff. auf allen vorliegenden Daten bestimmten Konfidenzwerte gezogen werden. Diese liegen bei Werten um 25% beziehungsweise bei der dezimierten Regelmenge bei 75%. Diese Werte können als repräsentativ angesehen werden, und als Grundlage

für die Konfidenzwerte der regelbasierten Präventionsmodelle angesehen werden. Anders ausgedrückt heißt das, daß auf jeden Fall bessere Werte mit Hilfe dieser Ansätze erzielt werden können, als auf der Regelfilterung alleine. Die in Abschnitt 3.12.4 referierten Werte bilden in diesem Zusammenhang eine „untere Schranke“ oder können als sogenannter *worst case* in diesem Zusammenhang angenommen werden.
Es ist weiterhin in diesem Zusammenhang zu beachten, daß sich die Konfidenz aufgrund des Hochrechnungsfaktors, der bei den falsch eingeordneten, legalen Datensätzen angewandt wird, schon auf sehr wenige solcher Fehlalarme reagiert und stark abfällt. Die Konfidenzfunktion ist vergleichbar mit der Inversfunktion $i(x) = \frac{1}{(h \cdot x)}$ mit h als Hochrechnungsfaktor. Der Konfidenzverlauf ist als gestrichelte Linie in Abbildung 6.7 dargestellt. Für die Ergebnisse des zweiten Ansatzes bedeutet das, daß die nur 3 Falschzuweisungen bezüglich der legalen Transaktionen aufgrund einer nur leichten Verschiebung der Klassengrenze (in Richtung `C` in Abbildung 6.7) verantwortlich für die extrem geringe Konfidenz sind. Gleiches gilt auch für die regelbasierten Ansätze, bei denen nur ein Fehlalarm zu einer Konfidenz von 50% führt. Ebenfalls damit verbunden ist in der Regel eine Steigerung der Mißbrauchstrefferquoten. Dies wird in Abbildung 6.7 daraus erkenntlich, daß bei einer Verschiebung der Klassengrenze in Richtung `C` nun ein größerer Teil der Trefferquotenkurve bezüglich der Mißbrauchszuweisungen auf der linken Seite der Klassengrenze zu liegen kommt.

Das abschließende, hierarchische Modell ist nun insofern auf eine Optimierung der Konfidenz ausgerichtet, als daß durch die primäre Regelfilterung die Klassengrenze am Beispiel der Abbildung 6.7 maximal in Richtung `B` verschoben wird. Anschließend wird diese dann durch die Analogdatenauswertung im Zweifelsfall eines Mißbrauchs in Richtung `C` korrigiert, ohne dabei einen Konfidenzverlust aufgrund falsch zugewiesener Mißbräuche zu riskieren. Eine Korrektur der sehr guten Zuordnung der legalen Datensätze durch die Regelfilterung ist nicht möglich (Abbildung 6.6). Es ist nur im Falle der Mißbrauchszuweisung eine Korrektur zugelassen. Auf diese Weise bleibt die Konfidenz auf Basis der Regelfilterung durch die generalisierten Regeln zu einem großen Teil unbeeinflußt von der potentiellen Mißbrauchsentscheidung durch die neuronalen Netzwerke.
Anders verhält es sich in diesem Zusammenhang bei dem ersten, regelbasierten Modell sowie auch mit dem parallelen Auswertungsmodell aus Abschnitt 6.3. Bei ersterem ist durch die sekundäre Klassifizierung auf Basis der Analogdaten eine Korrektur sowohl bei einer Klassenentscheidung in Mißbrauch oder in nicht Mißbrauch zu beiden Seiten möglich (Abbildung 6.5). So kann sowohl bei einer Mißbrauchsentscheidung eine Korrektur in Richtung `C` am Beispiel der Abbildung 6.7, als auch bei einer Entscheidung bezüglich einer legalen Transaktion in Richtung `B` durchgeführt werden. Es kann also unter Umständen trotz einer regelbasierten Entscheidung zugunsten der legalen Klasse, zu einer Mißbrauchszuordnung kommen. Dieses Faktum ist auch bei dem zweiten, dem konkurrierenden

Modell gegeben, wo eine einheitliche Entscheidung der analogdatenauswertenden Netze entgegen der Regelfilterung zu einer solchen Korrektur führen kann. Von diesen Eingriffen in die regelbasierte Entscheidung bezüglich der legalen Daten ist die Konfidenz jedoch stark aufgrund der Hochrechnung betroffen, so daß es seitens dieser Einstufung zu einem extremen Verlust kommt.

6.5.2 Mißbrauchstrefferquoten

Generell können bei den verschiedenen Ansätzen Trefferquoten bezüglich der Zuweisung der Mißbrauchsdaten von rund 80% erreicht werden. Bei dem regelspezifischen Ansatz aus Sektion 6.2 kommt es jedoch zu starken Schwankungen dieses Wertes, je nach Trainingsmethode. Die besten Ergebnisse können bei dem speziell trainierten Netzen erreicht werden. Allgemein läßt sich zu der regelbasierten Auswertung bezüglich des Mißbrauchs sagen, daß je größer die beim Training vorliegende Mißbrauchstransaktionsmenge ist, desto besser kann Mißbrauch letztendlich aufgedeckt und abschließend richtig zugeordnet werden. Dies äußert sich auch in den Ergebnissen des unparitätischen Trainings im parallelen Klassifikationsmodell aus Abschnitt 6.3, wo eine Abnahme der Mißbrauchstrefferquote mit der Zunähme des Trainingsverhältnisses gegeben ist. Im allgemeinen Regelmodell aus Abschnitt 6.4 kann durch die abschließenden Klassifizierung durch die neuronalen Netze die durch die Verschiebung der Klassengrenze hin zu `B` verlorene Mißbrauchstrefferquote wieder hergestellt werden, ohne dabei die Fehlerquote seitens der legalen Datensätze zu verringern. Die abschließend hinzugefügte Profilauswertung kann weiterhin zu einem Zugewinn bei den Trefferquoten beitragen, so daß der Fehleranteil des ergänzenden Klassifizierung auf ein Minimum reduziert werden kann.

6.5.3 Aufwandsabschätzung

Betrachtet man den Aufwand, der für die Klassifizierung der einzelnen Datensätze durchgeführt werden muß, so haben die regelbasierten Ansätze den Vorteil mit der Regelfilterung den Großteil der Datensätze, die potentiell legalen, vor einer weiteren Analyse herausfiltern zu können. Diese Filterung kann in beiden Fällen bei einer Regelmenge von ungefähr nur 700 Regeln sehr schnell ausgeführt werden. Im Gegensatz dazu wird für eine Klassifizierung wie im Fall des parallelen Klassifikationsansatzes aus Abschnitt 6.3 eine vollständige Abarbeitung *aller* Analysemethoden erwartet, um eine Entscheidung treffen zu können. Jedoch ist es möglich die einzelnen Teilentscheidungen nahezu parallel entgegen den den übrigen Analysemethoden auszuführen, so daß bezüglich der Laufzeit kein nennenswerter Nachteil entstehen kann.
Der regelbasierte Ansatz, der speziell die einzelnen Regeln mit eigenen neuronalen Netzwerken ausstattet, weist jedoch einen hohen Wartungs- und Trainings-

aufwand auf, da die einzelnen Netzparameter mitgeführt und gepflegt werden müssen. Es ist jedoch eine genauere regelspezifische Analyse der Analogdaten beziehungsweise des Profils möglich, die, wie sich zeigt, jedoch zu keinem erkennbaren Vorteil gegenüber dem allgemeinen, regelspezifischen Ansatz führt.
Also auch bezüglich der Wartbarkeit oder der Aktualisierbarkeit der Analysemethoden kann das abschließende, regelbasierte Analysemodell favorisiert werden; sind hier nur die Parameter und Gewichte *einer* neuronalen Netzinstanz im Gegensatz zum Ansatz aus Sektion 6.2 zu pflegen.

Kapitel 7

Resümee

Die Versuche haben gezeigt, daß mit den vorgestellten Analysemodellen zufriedenstellende Ergebnisse bezüglich der Mißbrauchsprävention erzielt werden können. Es konnten Verfahren entwickelt werden, die eine Alternative zur automatisierten Beurteilung von Transaktionen, wie sie im Kreditkartenbereich anfallen, darstellen. Die Analysemodelle ermöglichen eine auf stochastisch, statistischen Verfahren basierende, dynamische Mißbrauchserkennung bei den Autorisierungsverfahren, die die Zielsetzung einer möglichst schnellen und sicheren Verarbeitung gewährleisten.

7.1 Zusammenfassung

Ziel war es, eine automatische Auswertung der Transaktionsdaten durchzuführen, um mit wenigen gezielten Maßnahmen Mißbrauchstransaktionen zu erkennen und abwehren zu können.
Als Grundlage der präventorischen Analyse der Transaktionsdaten wurde ein auf streng statistischen Grundlagen basierendes Assoziationsmodell wie in Abschnitt 3.4 verwendet. Um den zugrundeliegenden Mißbrauchsraum einzuschränken und die Vergleichsroutine zu beschleunigen, wurden die einzelnen Mißbrauchsregeln – die Wertetupel, bestehend aus den symbolischen Daten – wie in Abschnitt 3.9 beschrieben, generalisiert. Mit der so generierten Regelmenge ist es nun möglich, die gewünschten Assoziationen aus Regel und Mißbrauchsart schnell durchführen zu können. Der Regelumfang konnte durch diese Generalisierungsmaßnahme auf weniger als ein Zehntel der eigentlich zugrundeliegenden Mißbrauchsregelmenge dezimiert werden, ohne daß in starkem Maße die Zuordnungssicherheit davon im negativen Sinne beeinflußt wird (Abschnitt 3.12).
Eine dynamische Anpassung kann ebenfalls durch eine laufende Korrektur der Statistikwerte bezüglich der einzelnen Regeln durchgeführt werden, so daß dieser Part der Analyse auf einem weitestgehend, aktuellen Stand gehalten werden kann.

Als unterstützende, die regelbasierten Entscheidungen beurteilende Maßnahme, konnten trainierte, neuronale Netze hinzugezogen werden. Mit Hilfe dieser Analysemethoden ist es möglich die analogen Daten bei der Übermittlung der Transaktion auszuwerten. Es hat sich jedoch gezeigt, daß eine alleinige Analyse und Beurteilung der Kreditanfragen mit Hilfe dieser neuronalen Netzarchitekturen ohne weitere Analysemethoden nicht für die gegebene Problemstellung geeignet ist. Dies ist insofern auch leicht einsichtig, als daß es für ein derartig trainiertes, neuronales Netz es nicht möglich ist, eine Erkennung bei sich überlappenden Mustern von über 99,9% zu erreichen.
Gemeinsam mit der auf rein statistischen Verfahren basierenden Auswertung der symbolischen Daten können jedoch anschließend oder parallel dazu mit Hilfe dieser Netzarchitekturen erfolgreiche Zugewinne seitens der Mißbrauchstrefferquote und der Zuordnungssicherheit bezüglich der Mißbrauchszuweisung erzielt werden. Dies zeigen auch die Ergebnisse in den einzelnen Abschnitten in Kapitel 6.
Erwähnenswert ist in diesem Zusammenhang, daß bis zu diesem Zeitpunkt der Auswertung sämtliche erzielten Ergebnisse ohne Berücksichtigung der zeitlichen Abfolge der Transaktionen sowie der kontenspezifischen Eigenschaften und Statistiken in Form von Benutzerprofilen durchgeführt wurden. Sämtliche Ergebnisse sind *komplett* benutzer- und zeitunabhängig bestimmt worden!
Eine weitergehende Analyse durch Profile und Zeitreihen birgt aureichend Potential, um zu einer Steigerung der Erkennungsquote beitragen zu können, wie in den Abschnitten 6.3 und 6.4 ersichtlich wird. Bei den derart ermittelten Ergebnissen ist zu berücksichtigen, daß es sich erneut um kontounabhängige Auswertungen handelt. Es ist zu vermuten, daß die Ergebnisse bezüglich der Trefferquoten und der Konfidenz noch einmal verbessert werden können, wenn ein streng karteninhaberorientiertes Benutzerprofil in die Analyse mit einbezogen wird. Kann eine Transaktionshistorie in ausreichend schneller Zeit ermittelt werden, so stehen sämtliche Mittel zur Verfügung, eine optimale Auswertung diesbezüglich durchzuführen, so daß Verhaltensweisen wie das „Abräumprofil“ sicher aufgedeckt werden können. Bei den zeitabhängigen Transaktionsauswertungen ist zu beachten, daß auf den zugrundeliegenden Daten keine aus kontinuierlichen Transaktionsfolgen bestehenden Zeitsequenzen und -reihen abgeleitet werden können, da es sich nur um eine Auswahl an Transaktionsdaten handelt. Dies führt wie angesprochen unter Umständen zu einer Verfälschung der Ergebnisse, die auf diese Art und Weise erzielt werden. Dieses Faktum fließt also erschwerend bei der simulierenden Klassifikation auf Basis der zeitabhängigen Transaktionsdaten mit ein.

Ein wesentlicher Punkt im Zusammenhang mit den vorgestellten Auswertungsmodellen und den damit erzielten Ergebnissen ist, daß diese auf einer Datengrundlage angewandt und erzielt wurden, die im voraus schon durch Expertensysteme von Mißbrauchstransaktionen bereinigt und analysiert wurden. Ein Großteil des eigentlichen Mißbrauchs wurde also durch diese Auswertung erkannt und frühzei-

tig verhindert. Die dennoch guten Ergebnisse können aufgrund dessen besonders hervorgehoben werden, fand die Analyse doch quasi unter „erschwerten Bedingungen“ statt.

Abschließend läßt sich das zuletzt vorgestellte, hierarchisch gestaffelte Regelmodell auf Basis einer Regelmenge (Abschnitt 6.4) für die vorgegebene Problemstellung empfehlen. Es können hier die besten Ergebnisse erzielt werden, sowie gestaltet sich die Verarbeitung der Autorisierungsanfragen in der selektierenden Auswertungsreihenfolge am übersichtlichsten und gut nachvollziehbar. Durch eine primäre Regelfilterung erübrigt sich für einen Großteil der Transaktionsdatensätze die Analyse durch die analogdatenauswertenden Schichten, so daß eine Autorisierung nur im Zweifelsfall durch diese Art der Analyse verzögert wird.
Interessant in diesem Zusammenhang ist, daß der Umfang der tatsächlich eingesetzten Expertenregeln, die zur Zeit bei der Mißbrauchsanalyse im Kreditinstitut eingesetzt werden, in etwa dem der automatisch erzeugten Regeln entspricht. Sicher können einige Regeln durch eine zielgerichtete Kombinatorik mit dem Hintergrund der Mißbrauchsalternativen manuell, optimiert gebildet werden. Dennoch ist in der Symbiose aus einer automatischen Regelerzeugung und mit dem Wissen von Experten auf jeden Fall ein sicherer Weg gegeben, die Mißbrauchsanalyse weiter zu verbessern.

7.2 Ausblick

Trotz der beschriebenen Erfolge bei der Mißbrauchsanalyse der Transaktionsdaten, sind weitere Alternativen zur Verbesserungen denkbar. Eine entsprechende Implementierung und Simulation scheiterte diesbezüglich am Umfang dieser Arbeit beziehungsweise an den eingeschränkten Mitteln, die zur Simulation zur Verfügung standen.

So ist es denkbar, den Generalisierungsalgorithmus aus Kapitel 3 weiter zu optimieren. In diesem Zusammenhang ist es denkbar, die zugrundeliegenden Daten weiter zu analysieren, beziehungsweise mit Hilfe von kreditinstitutsinternem Hintergrundwissen zu perfektionieren, und mit Hilfe eines weiter angepaßten Algorithmus', so wie es im Zuge der Entropie und der Vergleichsreihenfolge vorgenommen wurde, die Generalisierung zu beschleunigen.

Desweiteren ist eine weitergehende Optimierung der RBF-Netze denkbar, wie schon im eigentlichen Kapitel 4 angeschnitten. Zu einer Verbesserung der Auswertung der analogen Daten können optimierte Trainingsverfahren auf größeren Trainingsmengen, die in der Realität sicher vorliegen, beitragen, sowie angepaßtere radiale Basisfunktionen. Es ist weiter denkbar, zu den vorverarbeitenden Filtern eine weitere Schicht einzufügen, die eine grobe „Vorabklassifizierung“,

zum Beispiel durch eine *Principal Component Analyse* (PCA) oder ähnliches, durchführt.
Auch weitere Unternetze sind in dem Zusammenhang der analogen Daten denkbar. Ein Netztyp, der aus mangelnden Informationen bezüglich der Transaktionsdaten scheiterte, besteht aus der Auswertung von Zeit und Lokalität der getätigten Transaktion. Mit Hilfe dieser Kombination an Daten ist es leicht möglich, im Ausland geklonte Karten aufzuspüren, da Transaktionen innerhalb kurzer Zeit an Orten, die über große Entfernungen getrennt sind, damit erkennbar gemacht werden können.

Besonders auf Seiten der Benutzerprofile und Zeitreihen- beziehungsweise Zeitsequenzanalysen sind einige Verbesserungen denkbar, die jedoch unter anderem eine größere Datenbasis sowie eine ausgebaute Datenverwaltung und -logistik benötigen. Dabei sollte unter anderem berücksichtigt werden, daß kontinuierliche Transaktionshistorien zur Analyse verwendet werden, um das Ergebnis verfälschende „Zeitlöcher“ zu verhindern. Ein weiterer, wesentlicher Punkt sind zum einen statistische Benutzerprofile, mit deren Hilfe es möglich ist, das Verhalten der Karteninhaber vergleichend bei der Mißbrauchsanalyse einzusetzen, wie auch eine Auswertung der unmittelbar zurückliegenden Transaktionshistorie. Mit letzterem ist es möglich, spezialisierter als in Kapitel 5 vorgestellt, Transaktionsfolgen und Verhaltensweisen wie das „Abräumverhalten“ zu erkennen und für die Mißbrauchsanalyse zu verwenden.

7.3 Abschluß

Abschließend kann eine Analyse der Transaktionsdaten auf Grundlage der vorgestellten Verfahren sehr empfohlen werden. Es ist damit möglich, dynamisch die Transaktionsanalyse an die veränderten Mißbrauchsverhalten anzupassen, und damit die Mißbrauchserkennung mit Hinblick auf die Zuordnungssicherheit zu optimieren.

Literaturverzeichnis

[AS94] Agrawal, Rakesh und Ramakrishnan Srikant: *Fast Algorithms for Mining Association Rules.* Proceedings of the 20th VLDB Conference Santiago, 1994.
http://www.almaden.ibm.com/cs/people/ragrawal/pubs.html#associations [Stand April 1999].

[BG98] Brand, Estelle und Rob Gerritsen: *Naïve Bayes and Nearest Neighbor.* DBMS- Online, 1998.
http://www.dbmsmag.com/9807m07.html [Stand April 1999].

[BLH99] Brause, Rüdiger, Timm Langsdorf und Hanns-Michael Hepp: *Neuronal Data Mining for Credit Card Fraud Detection. submitted to* IEEE Int. Conference Of Tools With Artificial Intelligence, 1999.

[Bol96] Bollinger, T.: *Assoziationsregeln – Analyse eines Data Mining Verfahrens.* Informatik - Spekturm, (19):257 – 261, 1996.

[Bra95] Brause, Rüdiger: *Neuronale Netze.* B.G. Teubner Stuttgart, 1995. 2. Auflage.

[BSMM93] Bronstein, Ilja N., Konstantin A. Semendjajew, Gehard Musiol und Heiner Mühlig: *Taschenbuch der Mathematik.* Verlag Harri Deutsch, 1993.

[CE92] Cardaliaguet, Pierre und Guillaume Euvrard: *Approximation of a Function and its Derivative with a Neural Network.* Neural Networks, 5:207–220, 1992.

[Cov65] Cover, T.M.: *Geometrical and statistical properties of systems of linear inequalities with applications in pattern recognition.* IEEE Transactions on Electronic Computers, EC-14:326–334, 1965.

[Ea93] Engesser, Hermann und andere: *Duden »Informatik«.* Dudenverlag, 1993.

[Fla98] Flanagan, David: Java *In A Nutshell.* O'Reilly Verlag, 1998.

[FP97] Fawcett, Tom und Foster Provost: *Adaptive Fraud Detection.* Data Mining and Knowledge Discovery, (1):291–316, 1997.

[Gem99] Gema, Raoul: *Vielfalt für* Java . c't magazin für computer technik, 9:30, 1999.

[Hay94] Haykin, Simon: *Neural Networks.* Prentice Hall, 1994.

[HH98] Hilderman, Robert J. und Howard J. Hamilton: *Mining Association Rules From Market Basket Data Using Share Measures and Characterized Itemsets.* International Journal on Artificial Intelligence Tools, 7(2):189–220, 1998.

[Kim97] Kimball, Ralph: *Preparing For Data Mining.* DBMS- Online, 1997.
http://www.dbmsmag.com/9711d5.html [Stand April 1999].

[MD89] Moody, J. und C. Darken: *Learning with Localized Receptive Fields.* Touretzky et al., Seiten 133 – 143, 1989.

[OW90] Ottmann, Thomas und Peter Widmayer: *Algorithmen und Datenstrukturen.* BI-Wissenschaftsverlag, 1990.

[Pat97] Patterson, Dan: *Künstliche neuronale Netze.* Prentice Hall, 1997. 2. Auflage.

[Pfl86] Pflug, Georg: *Stochastische Modelle in der Informatik.* B.G. Teubner Stuttgart, 1986.

[Pie95] Pietruschka, Ulf: *Funktionsapproximation mit RBF- Netzen.* Diplomarbeit, J.W. Goethe Universität, 1995.

[Roj93] Rojas, Raul: *Theorie der neuronalen Netze.* Springer-Verlag, 1993.

[SA95] Srikant, Ramakrishnan und Rakesh Agrawal: *Mining Generalized Association Rules.* Proceedings of the 21st VLDB Conference, 1995.

[Toi] Toivonen, Hannu et al.: *Pruning and Grouping Discovered Association Rules.*
http://www.cs.helsinki.fi/research/fdk/datamining/pubs/
crete95.ps.gz [Stand April 1999].

[XKO93] Xu, Lei, A. Krzyzak und E. Oja: *Rival Penalized Competitive Learning for Clustering Analysis, RBF Net, and Curve Detection.* IEEE Transactions on Neural Networks, 4(4):636–649, 6 1993.

[Zel97] Zell, A.: *Simulation Neuronaler Netze.* Oldenburg, 1997.

Anhang A

Datenbeschreibung

Abkürzung	Beschreibung	Format
ACCT_NBR	Alphanumerisch verschlüsselte Kartennummer	symbolisch
TRN_DT	Transaktions Datum im Format: 'YYYY.MM.DD HH:MM:SS'	analog
TRN_TYP	Transaktionstyp, 3-stelliger Kode: 1.Stelle: Acquiring Typ 2.Stelle: Finanzieller Bezug 3.Stelle: Transaktionsbezug	symbolisch
TRN_AMT	Transaktionsbetrag in Pfennigen im Format: '#,00' (trotzdem Pfennige)	analog
AVAL_BALLNCE	Verfügungsrahmen zum Zeitpunkt vor der Transaktion in Pfennigen im Format: '#,00' (trotzdem Pfennige) „Wie weit darf das Konto noch überzogen werden?“	analog
CURR_CD	Währungskode, nummerischer ISO-Währungsschlüssel	symbolisch
POS_ENT_CD	**P**oint **O**f **S**ale, 12 stelliger Code	symbolisch

Fortsetzung auf nächster Seite

Abkürzung	Beschreibung	Format
FAL_SCOR	FALCON-Score, durch ein neuronales Netz erzeugte Bewertung. Eine 3 stellige analoge Zahl im Intervall [0 ... 999]	symbolisch
START_DT	Karte gültig ab; Datumsformat: 'YYMM'	analog
EXP_DT	Karte gültig bis; Datumsformat: 'YYMM'; entspricht dem Verfalldatum	analog
CR_LMT	Kreditrahmen in Pfennigen im Format: '#,00' (trotzdem Pfennige)	analog
CRD_TYP	Kartentyp: EM = Eurocard/Mastercard sonst unbekannt	symbolisch
ICA_CD	**I**nterbank **C**ard **A**ccessnumber (Transferpartner); jedoch **nur** außerhalb von Europa vergeben!	symbolisch
AID_CD		symbolisch
SIC_CD	Branchenschlüssel; Kodierung der Branchenbezeichnung, wo die Transaktion ausgeführt wurde (z.B. Tankstelle, Bank, Kaufhaus, etc.)	symbolisch
ACT_CD	GANS Action Code; Authorisierungsresponse	symbolisch
MSG_TYP	entweder *Anfrage*- oder *Antwort*typen	symbolisch
MER_ID	Händlernummer, Nummer des Vertragsunternehmen	symbolisch
MER_CNTY_CD	Länderkennzeichen des Vertragsunternehmens	symbolisch

Tabelle A.1: Transaktionsdatensatz

Abkürzung	Beschreibung	Format
ACCT_NBR	Alphanumerisch verschlüsselte Kartennummer	symbolisch
MER_CTY	Ort des Vertragsunternehmen	symbolisch
MER_POST_CD	Postleitzahl des Vertragsunternehmens	symbolisch
MER_ST	Straße des Vertragsunternehmens	symbolisch
TRN_AMT	Transaktionsbetrag im Format: '#,##' (diesmal DM-Beträge)	analog
TRN_DT	Transaktions Datum im Format: 'YYYY.MM.DD HH:MM:SS'	analog
TRN_ORG	Transaktionsherkunft: 0 = Deutschland 1 = Europa 2 = Andere	symbolisch
TRN_CNTY_CD	Länderkennzeichen des Landes, aus dem die Autorisierungsanfrage kam	symbolisch
MER_ID	Händlernummer, Nummer des Vertragsunternehmen	symbolisch
SIC_CD	Branchenschlüssel; Kodierung der Branchenbezeichnung, wo die Transaktion ausgeführt wurde (z.B. Tankstelle, Bank, Kaufhaus, etc.)	symbolisch
FRD_TYP	Interne Mißbrauchsbeschreibung	symbolisch
ICA_CD	**I**nterbank **C**ard **A**ccessnumber (Transferpartner); jedoch **nur** außerhalb von Europa vergeben!	symbolisch

Tabelle A.2: Ergänzende Mißbrauchsdaten zu den Transaktionendaten aus A.1

Abkürzung	Beschreibung	Format
ACCT_NBR	Alphanumerisch verschlüsselte Kartennummer	symbolisch
CTY_1	Ort der Wohnadresse	symbolisch
POST_CD_1	Postleitzahl der Wohnadresse	symbolisch
CNTY_CD_1	Länderkennzeichen der Wohnadresse	symbolisch
BIRTH_DT	Geburtsdatum des Karteninhabers im Format: 'YYYYMMDD'	analog
EXP_DT	Kartenverfallsdatum im Format: 'YYYYMMDD'	analog
OPN_DT	Karteneröffnungsdatum im Format: 'YYYYMMDD'	analog
CR_LMT	Karten Kreditlimit in Pfennigen	analog
ACTV_IND	Aktiv- Kennzeichen: A = Karte aktiv I = Karte inaktiv	symbolisch
ACCT_STAT	Sperrgrund	symbolisch
CTY_2	Ort der Versandadresse	symbolisch
POST_CD_2	Postleitzahl der Versandadresse	symbolisch
ADDR_STAT	Addressenstatus	symbolisch
ISS_DT	Datum des letzten Kartenversands im Format: 'YYYYMMDD'	analog
EMIT_NBR	Emittenten Nummer, Nummer des Kartenvertreibers	symbolisch
INST_NBR	Instituts Nummer, Nummer des kartenvertreibenden Instituts	symbolisch
ISS_REAS	Versandgrund: N = Neukarte A = Austausch wegen defekter Karte T = Austausch wegen Kartenverlust R = normale Austauschkarte I = Reaktivierung	symbolisch
GEN_CD	Anredeschlüssel: 000 = Keine Anrede 001 = Herr 002 = Frau 003 = Fräulein 004 = Firma	symbolisch
CARD_TYP	Kartentyp: S = Standardkarte G = Goldkarte H = Mastercard Standardcard I = Mastercard Goldkarte K = Firmenkunden Standardkarte L = Firmenkunden Goldkarte	symbolisch

Tabelle A.3: Karteninhaberdatensatz

Anhang B

Ergebnisse der Datenanalyse

Feld	Min	Max	Kommentar
illegale Transaktionen bzw. gesperrte Konten			
TRN_DT	1997.05.17 07:44:24	1998.01.19 21:45:35	
TRN_AMT	5,00	920116,00	in Pfennigen!
AVAL_BALLNCE	0,00	5381166,00	in Pfennigen!
START_DT	0000	0000	s.u.
EXP_DT	0000	9912	s.u.
CR_LMT	0,00	0,00	s.u.
BIRTH_DT	19000101	19810108	
EXP_DT	19970630	20001130	
OPN_DT	19890215	19971110	
CR_LMT	0,00	800000,00	
legale Transaktionen bzw. aktive Konten			
TRN_DT	1997.05.17 00:01:45	1998.04.07 09:28:47	
TRN_AMT	0,00	99999,00	in Pfennigen!
AVAL_BALLNCE	0,00	999999,00	in Pfennigen!
START_DT	0000	0000	s.u.
EXP_DT	0000	9912	s.u.
CR_LMT	0,00	5000000,00	s.u.
BIRTH_DT	09760307	19971223	
EXP_DT	19950430	20010430	
OPN_DT	19790108	19980327	
CR_LMT	0,00	9900000,00	

Tabelle B.1: Statistische Auswertung der analogen Daten getrennt für legale beziehungsweise illegale Transaktionsdaten und aktive und gesperrte Konten

Feld	Anzahl	Kommentar
Transaktionsdaten		
ACCT_NBR	43902	
TRN_TYP	27	
CURR_CD	40	
POS_ENT_CD	81	
FAL_SCOR	918	
CRD_TYP	2	entweder `null` oder `EM`
ICA_CD	699	
AID_CD	2033	
SIC_CD	468	
ACT_CD	49	
MSG_TYP	7	
MER_ID	138765	
MER_CNTY_CD	1	immer `null`
Karteninhaberdaten		
CTY_1	6601	
POST_CD_1	6690	
CNTY_CD_1	74	
CR_LMT	37	
ACTV_IND	2	nur die Werte `A` & `I`
ACCT_STAT	12	
CTY_2	2617	
POST_CD_2	3570	
ADDR_STAT	2	`null` und `FL`
EMIT_NBR	2778	
INST_NBR	4587	
ISS_REAS	7	
GEN_CD	4	
CARD_TYP	6	

Tabelle B.2: Statistische Auswertung der symbolischen Daten auf allen Datensätzen (Wertestreuung)

Anhang C

Der Generalisierungsalgorithmus

Im folgenden ist der Generalisierungsalgorithmus stark vereinfacht in Form von Pseudocode dargestellt. Man erkennt deutlich die starke Schachtelung der einzelnen Iterationsschleifen.

Algorithmus C.1

```
1  while (currList.length() > 0) ∧ (ruleDist < ROWLEN) do
2        for i := 1 to currList.length() do
3            for j := i + 1 to currList.length() do
4                if (ruleDiff(rule_i, rule_j) = ruleDist)
5                  then workRule := mergeRule(rule_i, rule_j);
6                       if (lookUp(workRule, currList, newList) = false)
7                         then confidence := calcConf(workRule);
8                              share := calcShare(workRule);
9                              if (confidence > minConf)
10                               then insert(workRule, newList);
11                                    mark(rule_i);
12                                    mark(rule_j);
13                                else if share > minShare
14                                       then workRule := searchSubTree(rule_i, rule_j);
15                                            if (lookUp(workRule, currList, newList) = false)
16                                              then insert(workRule, newList);
17                                                   mark(rule_i);
18                                               else workRule := searchSubTree(rule_j, rule_i);
19                                                    if (lookUp(workRule, currList, newList) = false)
20                                                      then insert(workRule, newList);
21                                                           mark(rule_j);
22                                                    fi
23                                            fi
24                                   fi
25                             fi
26                      fi
27               fi
28           od
29       od
30       if (newList.length() == 0)
```

```
31          then ruleDist := ruleDist + 1;
32           else ruleDist := 0;
33        fi
34        deleteMarked(currList);
35        currList := concatenate(currList, newList);
36        writeOut(newList);
37  od
```

Variablenname	Beschreibung
$currList$	aktuelle Arbeitsliste
$newList$	Liste der neu gefundenen Regeln
$rule_x$	Regel x in der Arbeits- Regelliste
$ROWLEN$	Regellänge, Anzahl der Items (z.B. 27)
$ruleDist$	akt. Regeldistanz (initial 0)
$minConf$	Mindestkonfidenz
$minShare$	Mindestabdeckung

Tabelle C.1: Beschreibung der in Algorithmus C.1 benutzten Variablen

Methodenname	Beschreibung
ruleDiff()	Bestimmt die Anzahl der Unterschiede zweier Regeln. Liegt die Entropie der verglichenen Elemente unter *min-Entropie* und ist ein Wildcard in dem Vergleich beteiligt, dann wird die Wildcardliste durchsucht, sonst kommt es zu **keinem** Unterschied!
mergeRule()	Fügt zwei Regeln zu einer neuen zusammen, wobei an den Stellen, wo sich die einzelnen Elemente unterscheiden, ein Wildcard eingefügt wird. Im Falle, daß die Entropie des Merkmals unter der *min-Entropie* liegt, werden die verschiedenen Elemente in eine Wildcardliste aufgenommen.
lookUp()	Sucht nach dem Auftreten der Regel (1. Parameter) in den folgend angegebenen Listen (Parameter $2, \ldots, n$).
calcConf()/calcShare()	Berechnet Konfidenz beziehungsweise Abdeckung auf der Auswahl von legalen und illegalen Regeln.
insert()	Fügt die angegebene Regel in die Liste ein.

Fortsetzung auf nächster Seite

Methodenname	Beschreibung
mark()	Markiert die Regel, damit sie zu einem späteren Zeitpunkt aus der Liste der aktiven Regeln herausgenommen werden kann.
searchSubTree()	Sucht nach einer Regel mit ausreichender Konfidenz und passendem Regelabstand, zusammengefügt aus der aktuellen Regel (1. Parameter) und Regeln aus dem Unterbaum der 2. angegebenen Regel (2. Parameter). Diese Funktion wird rekursiv bis zu den Ursprungsregeln auf GL 0 aufgerufen.
deleteMarked()	Löscht die markierten Regeln aus der angegebenen Liste, wobei sie jedoch für eine eventuell Baumsuche (s.o. searchSubTree()) im Speicher belassen werden.
concatenate()	Fügt zwei Listen zusammen. Redundante Regeln werden entfernt.
writeOut()	Schreibt die Ergebnisse heraus.
length()	Gibt die Länge der Liste aus.

Tabelle C.2: Funktionserläuterungen für Kodebeispiel C.1

Anhang D

Klassenhierachie der RBF-Netzimplementierung

An dieser Stelle soll grob auf die Implementierung der RBF-Netzarchitektur eingegangen werden. Dazu wird in kurzer Form die Klassenhierarchie der verwendeten Klassen beschrieben und die Beziehungen unter den einzelnen Klassen erläutert. Abbildung D.1 macht die Zusammenhänge in graphischer Form deutlich. Die Oberklasse oder Superklasse wird durch ein sogenanntes *Interface*, ein

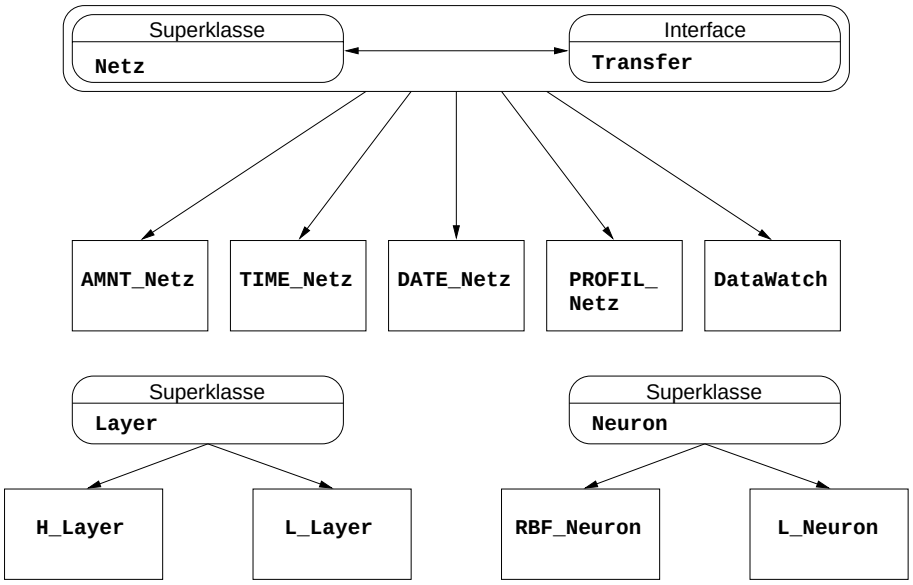


Abbildung D.1: Klassenhierarchie der RBF-Netzimplementierung

JAVA spezifisches Konstrukt, repräsentiert. Bei JAVA kann eine Klasse eigentliche Implementierungen nur von *einer* Superklasse erben. Sie kann zusätzlich `abstract`-Methoden von Interfaces erben, muß aber ihre eigenen Implementierungen dieser Methoden bereitstellen. Auf diese Weise ist es nun möglich, die

Grundstruktur eines Netzes in der Klasse `Netz` zu implementieren, und die notwendigen, aber speziellen Filtermethoden in den einzelnen Netzinstanzen zu realisieren. Die `abstract`-Methoden im Interface `Transfer` sehen diese Methoden nur vor, ohne sie zu implementieren, sorgen jedoch dafür, daß jede Instanz auf die zum Teil für sich spezifischen Methoden zugreifen kann. Trotzdem ist es auf diese Weise möglich, die einzelnen, spezialisierten Instanzen als eine zwar in manchen Methoden sich unterscheidende aber sonst einheitliche von Netz geerbte Instanzen zu behandeln.

Im folgenden werden nun die einzelnen Klassen beschrieben:

`Transfer`: Superklasse, die die grundlegenden und notwendigen Netzmethoden für alle spezialisierten Netzinstanzen vorsieht. Darunter fallen unter anderem `addInput()`, `getOutput` aber auch die Methode `getFillStatus()`, die für die Überwachung des Transaktionspuffers zuständig ist.

`Netz`: In dieser Superklasse sind die grundlegenden Methoden zur Ausführung der Klassifizierung mit Hilfe der RBF-Netze implementiert. Es wird in dieser Klasse das Training (`trainNet()`) sowie die Verifikationsphase (`workNet()`) näher spezifiziert, als auch die Verwaltung der grundlegenden Netzparameter.

`AMNT_Netz`: In dieser abgeerbten Klasse von Netz werden die `abstract`-Methoden der Superklasse `Transfer` mit Hinblick auf die Verarbeitung von Finanzdaten hin implementiert. Dazu zählt die Verrechnung, Normierung und Pufferung der relevanten Daten. Außerdem findet hier die Verrechnung der Relationen aus Kreditlimit als auch Kreditrahmen zum aktuellen Transaktionsbetrag.

`TIME_Netz`: Hier sind die `abstract`-Methoden darauf ausgelegt, die Transaktionsuhrzeit aus dem Datums-/Zeitfeld zu extrahieren und für das Netz aufzubereiten.

`DATE_Netz`: In dieser geerbten Klasse von Netz sind die Methoden auf die Verarbeitung von Datumswerten ausgelegt. Dazu zählt die Berechnung des Alters sowie die Bildung der Zeitdifferenz zwischen Eröffnungs- beziehungsweise Ablaufdatum der Karte und dem eigentlichen Transaktionsdatums.

`Profil_Netz`: Die Klasse `Profil_Netz` implementiert die notwendigen Netzmethoden in Hinblick auf die Analyse der Betragswerte in Zusammenhang mit den Zeitdifferenzen zwischen den einzelnen Transaktionen. Dazu ist eine Pufferung der Transaktionsdaten notwendig.

`DataWatch`: Hiebei handelt es sich im Prinzip nur um eine Dummy-Klasse, die es erlaubt, die gebildete Instanz genauso wie eine der anderen Netzinstanzen

zu behandeln. Nur wenige Methoden wie zum Beispiel die Ein- und Ausgabemethoden werden von den eigentlichen Netzmethoden weiterhin benutzt. Bei einem großen Teil der übrigen Netzmethoden handelt es sich in diesem Zusammenhang um Dummy-Methoden. Stattdessen wurde die Methode `checkSequence()` eingefügt, die die Überwachung der symbolischen Daten über eine bestimmte Anzahl an Transaktionen hinweg beobachtet. In diesem Zusammenhang müssen auch die beim Training ermittelten Wahrscheinlichkeitswerte (siehe Abschnitt 5.2.1) verwaltet, das heißt gespeichert und abgerufen werden können.

Die vollständig trainierten Netze werden zur Speicherung mit Hilfe der *Objektserialisierung*, einer Möglichkeit bei JAVA den kompletten Zustand eines Objekts (einschließlich aller Objekte, auf die es verweist) an einen Ein- oder Ausgabe-Stream zu übergeben, gespeichert. Damit kann bei Bedarf ein Training an dem Zustand, wo es abgebrochen wurde, ohne Einschränkung fortgeführt werden. Es werden auf diese Weise die Netzinstanzen in ihrem Orginalzustand zur späteren Verwendung gespeichert.

Weiter wurden die Superklassen `Layer` und `Neuron` implementiert. Es handelt sich hierbei auch wieder um abstrakte Klassen, die standardisierte Methoden der einzelnen Schichten und Neuronen vorsehen.

`Layer`: Diese Klasse verwaltet und steuert die Abarbeitung der einzelnen in ihnen enthaltenen Neurone und ist für die Datenübertragung an und von diesen Neuronen verantwortlich. Desweiteren können an dieser Stelle Zwischeninformationen ausgegeben werden, die auf die korrekte Arbeitsweise der einzelnen Schichten und Neurone schließen lassen.

- `H_Layer`: Die verdeckte Schicht, der sogenannte „hidden-layer“. Hier finden sich neben den `abstract`-Methoden wie zum Beispiel `setInput()` und `getOutput()` Methoden, die speziell auf die Arbeitsweise der RBF-Neurone abgestimmt sind. So wird hier die Verschiebung der Zentren sowie die Varianzänderung der RBF-Neurone initiiert. Auch wird in der Methode `calcMinDist()` die Entfernung zum nächsten Neuron beziehungsweise zum nächsten Zentrum einer Basisfunktion bestimmt.
- `L_Layer`: Die abschließende, die linearen Neuronen enthaltende Schicht. In dieser Schicht finden sich nur wenige, unmittelbar mit den linearen Neuronen in Zusammenhang stehende, spezielle Methoden. Generell sind hier nur die `abstract`-Methoden implementiert und auf die Verarbeitung der Daten für ein linerares Neuron ausgerichtet.

`Neuron`: Auch hier werden im wesentlichen die Ein- und Ausgabemethoden, und Informationen über den Zustand des Neurons vorgesehen. Alle übrigen, speziellen Methoden werden in den geerbten Klassen implementiert.

`RBF_Neuron`: Neben den `abstract`-Methoden sind hier im wesentlichen die Berechnungsmethoden der radialen Basisfunktionen untergebracht. Auch die eigentlichen Zentrumsverschiebungen beziehungsweise Varianzänderungen werden hier ausgeführt.

`L_Neuron`: Das Sigmaneuron beziehungsweise das lineare Neuron. Hier sind neben den Ein- und Ausgabemethoden hauptsächlich die im Zusammenhang mit der Delta-Lernregel benötigten Berechnungen implementiert. Außerdem werden hier die Gewichtsvektoren verwaltet, die die Eingaben kontrollieren.

Eine genaue Aufführung mit einer kurzen Beschreibung der einzelnen Methoden findet sich im Übrigen auf der beigefügten CD-Rom in Form einer von JavaDoc erzeugten Html-Dokumentation.

Anhang E

Entropiewerte der GZS Daten

Feld	Entropiewerte total	Entropiewerte illegal	Entropiewerte legal
ACCT_NBR †	-	-	-
TRN_TYP	1,28208	0,05346	1,28643
CURR_CD	1,41096	0,00307	1,41205
POS_ENT_CD	2,16257	2,18304	2,14545
FAL_SCOR	0,47076	0,99042	0,46104
CRD_TYP	0,00259	0,0	0,00262
ICA_CD	1,75954	4,12970	1,70476
AID_CD	3,05879	4,13224	3,01814
SIC_CD	3,75927	3,38992	3,74950
ACT_CD	0,26437	0,25809	0,26368
MSG_TYP	0,70239	0,02492	0,70373
MER_ID †	-	-	-
MER_CNTY_CD	0.00000	0.00000	0.00000
CTY_1	6,85155	5,37085	6,84784
POST_CD_1	8,00791	5,68897	8,01181
CNTY_CD_1	0,16628	0,77940	0,15521
CR_LMT	1,46739	0,96377	1,45765
ACTV_IND	0,05902	0,0	0,0
ACCT_STAT	0,15699	1,14883	0,08664
CTY_2	2,09774	2,37427	2,08357
POST_CD_2	2,54120	2,00415	2,53800
ADDR_STAT	0,00240	0,0	0,00242
EMIT_NBR	5,70722	4,64076	5,70964
INST_NBR	6,87843	5,45223	6,87910
ISS_REAS	0,83632	0,85713	0,83591
GEN_CD	0,90451	0,84909	0,90506
CARD_TYP	0,82468	0,86397	0,82413

Tabelle E.1: Entropiewerte

†Dieses Feld wurde nicht bearbeitet

Die Entropiewerte in der Tabelle sind entsprechend der Formel

$$H(x) = -\sum_i P_i \cdot \ln(P_i)$$

berechnet worden, wobei es sich bei P_i um die Auftrittswahrscheinlichkeit der einzelnen Merkmalsausprägungen handelt.

Es ist bei dem Feld `ACCT_IND` zu beachten, daß bei den illegalen beziehungsweise legalen Daten jeweils nur ein bestimmtes Datum (`I` oder `A`) auftaucht. Aufgrund der jeweiligen Wahrscheinlichkeit von 1 folgt daher für die Entropie ($\log(1) = 0$). Gleiches ist bei den Feldern `CARD_TYP` und `ADDR_STAT` auf Basis der illegalen Daten festzustellen.