DATA MINING
schatgraven in databases

Arno Siebes, Marcel Holsheimer

Centrum voor Wiskunde en Informatica
Postbus 94079, 1090 GB Amsterdam

1. Introduktie

Databases worden de laatste jaren steeds vaker gebruikt, de hoeveelheid informatie die in een database is opgeslagen neemt daarbij spectaculair toe. Een gevolg hiervan is dat het overzicht over de gegevens verloren gaat: waar het voor kleine hoeveelheden gegevens nog mogelijk was om globale veranderingen, trends en verbanden tussen gegevens te onderscheiden, wordt bij grote hoeveelheden gegevens de gebruiker het zicht hierop volledig ontnomen.

Een oplossing voor deze problemen is 'data mining', een techniek waarmee kennis kan worden gevonden in databases. Onder kennis verstaan we bijvoorbeeld (1) *verbanden* tussen kenmerken van objecten, zoals het verband tussen persoonskenmerken en kredietwaardigheid, en tussen objecten onderling, (2) *compacte beschrijvingen* op een hoog abstractie-nivo van de gegevens in een database, (3) en *trends* en globale veranderingen in de database.

Deze verbanden, beschrijvingen en trends vormen een bron van nuttige management informatie. In deze notitie gaan we in op de eerste vorm van data mining, het zoeken van verbanden in de database. We demonstreren aan de hand van voorbeelden wat data mining is en wat het nut van de met data mining verkregen informatie is. In het tweede, technischere, deel van deze notitie schetsen we hoe data mining in z'n werk gaat.

2. Wat is Data Mining?

Steeds meer bedrijven gebruiken steeds vaker databases bij de automatisering van hun bedrijfsprocessen. De hoeveelheid data die wereldwijd in databases is opgeslagen groeit exponentieel. Schattingen lopen uiteen van een verdubbeling per drie jaar tot een vertienvoudiging per vijf jaar.

Deze bergen gegevens vormen een ware schatkamer van informatie. Niet alleen de feitelijke opgeslagen informatie, maar ook informatie die daar uit afgeleid kan worden, zoals verbanden tussen gegevens. Neem bijvoorbeeld een hypothetische verzekeringsmaatschappij Hypos. Deze firma gebruikt al jaren een database om alle informatie omtrent autoverzekeringen op te slaan.

Dankzij deze databases is het voor Hypos nu heel eenvoudig om na te gaan hoeveel klanten een bepaalde verzekering hebben afgesloten, en hoeveel daarvan ooit schade hebben gereden. Maar de database bevat veel meer, impliciete, informatie zoals bijvoorbeeld *risico-profielen*. We kunnen op zoek gaan naar deze profielen, dus naar de persoonskenmerken die karakteristiek zijn voor klanten die schade hebben gereden.

Het vinden van zulke verborgen informatie heet *data mining*. Wat preciezer gezegd is data

mining het zoeken naar *karakteristieke patronen* in de database. Het zoeken naar risico-profielen is het zoeken naar een karakteristieke beschrijving van brokkenmakers.

Voor een concreter voorbeeld van data mining nemen we de volgende database:

Naam	Soort	Sterfelijk
Socrates	Mens	Ja
Plato	Mens	Ja
Heracles	Halfgod	Ja
Zeus	God	Nee
Bacchus	God	Nee

Karakteristiek voor de Goden in deze database is dat zij onsterfelijk zijn. Met andere woorden, met behulp van data-mining leiden we de regel af:

als sterfelijk = Nee dan soort = God

Als we karakteriseren wie er sterfelijk zijn, dan vinden we de regel:

als soort = Mens of soort = Halfgod dan sterfelijk = Ja

Zo'n uitdrukking 'soort = Mens of soort = Halfgod' noemen we een karakteristiek patroon voor een klasse, in dit geval de klasse 'sterfelijk'.

Data mining verloopt altijd in twee fasen. Eerst definïeren we de klasse, d.w.z. we geven aan welke eigenschap we willen karakteriseren, bijvoorbeeld sterfelijkheid. Hiermee verdelen we de database in twee groepen, zij die de eigenschap hebben en zij die de eigenschap niet hebben. Die in de eerste groep heten ook wel de positieve en die in de tweede de negatieve voorbeelden.

Vervolgens zoeken we naar een karakteristiek patroon voor de positieve voorbeelden. Dat wil zeggen, we zoeken naar een uitspraak die waar is voor alle positieve voorbeelden en onwaar voor alle negatieve voorbeelden.

Soms vinden we met data mining een regel die precies klopt, zoals dat onsterfelijkheid karakteristiek is voor goden. Soms vinden we alleen een statistische regel, zoals bij risico-profielen. Zo hebben jonge mannen een grotere kans op schade dan gemiddeld, maar niet alle jonge mannen krijgen rijden schade en niet iedereen met auto schade is een jonge man.

2.1 Waarom Data Mining?

De regels die met behulp van data mining gevonden worden geven inzicht in de samenhang en de inhoud van de database zonder dat deze daarvoor intensief bestudeerd hoeft te worden. Dit inzicht kan voor zowel strategische als controlerende taken worden gebruikt.

2.1.1 Strategisch Gebruik Het verzekeringsbedrijf Hypos kan aan de hand van de gevonden risico-profielen besluiten de premie van sommige groepen klanten te verhogen, terwijl de premie van andere groepen juist verlaagd wordt. Dit kan de concurrentie-positie verbeteren zonder dat de winstgevendheid in gevaar gebracht wordt.

Bij hypotheek-verstrekkers is er een soortgelijk strategisch gebruik mogelijk. Immers, het acceptatie-beleid van een bank is een verzameling regels die moeten worden toegepast om te beoordelen of een bepaalde klant een bepaalde hypotheek mag afsluiten. Deze regels zijn er

om het risico van de bank aanvaardbaar te maken. Er zijn echter toch gevallen waar de bank verlies lijdt omdat klanten hun verplichtingen niet na kunnen komen.

Met behulp van data mining kunnen risico-profielen opgesteld worden van de klanten die uiteindelijk tot verlies gaan leiden, deze regels kunnen gebruikt worden om het acceptatie-beleid bij te stellen.

2.1.2 Controle Het salaris van veel werknemers is in de CAO vastgelegd. Voor een, interne, accountant is het van belang om te controleren of de CAO-regels ook daadwerkelijk gehanteerd worden.

Een manier om dit te controleren is handmatig alle personeelsgegevens en functie-omschrijvingen te controleren. Zeker bij grotere bedrijven is dat een tijdsintensieve operatie waarvan het de vraag is of de kosten tegen de baten opwegen.

Data mining geeft een andere controlemogelijkheid. Zoek naar de karakteristieke patronen in de database die het salaris van een werknemer bepalen. Het resultaat zou bijvoorbeeld de volgende drie regels kunnen zijn:

als geboortedatum $>$ 1963 en salarisschaal $=$ 8 en afdeling $=$ produktie
dan salaris tussen 30.000 en 34.000
met waarschijnlijkheid 98% en geldend voor 64% van de database

als geboortedatum $>$ 1963 en salarisschaal $=$ 8 en afdeling $=$ verkoop
dan salaris tussen 35.000 en 38.000
met waarschijnlijkheid 100% en geldend voor 24% van de database

als geboortedatum $<$ 1963
of salarisschaal $=$ 9
dan salaris tussen 39.000 en 41.000
met waarschijnlijkheid 100% en geldend voor 12% van de database

Laten we even aannemen dat deze regels overeenstemmen met de CAO. Dan kan de accountant zich toespitsen op de uitzonderingen. De eerste regel heeft namelijk een dekking van 98%. Met andere woorden er zijn uitzonderingen, dat wil zeggen werknemers die na 1963 geboren zijn, in salarisschaal 8 zitten en bij de afdeling produktie werken, maar waarvan het salaris niet tussen de 30.000 en 34.000 ligt.

2.2 Data Mine Tools
De huidige generatie database management systemen, zoals DB2, Ingres en Oracle zijn niet geschikt om te data minen. De vraag: "Wat zijn de risico-profielen in mijn verzekeringsbestand?" kan er eenvoudigweg niet aan gesteld worden.

Handmatig zoeken is als alternatief niet realistisch: gezien de grootte van veel databases is het in één oogopslag zien van regelmaat ten ene male onmogelijk. De grootte van de database maakt ook het testen van alle mogelijke regels onmogelijk. Dit komt zowel door de grote hoeveelheid mogelijke manieren waarop de data gegroepeerd, oftewel *geclusterd*, kan worden als de grote hoeveelheid mogelijke patronen die voor een groep kunnen gelden.

Om dit te illustreren nemen we een database die de resultaten van een enquête bevat. Stel dat de enquête 10 vragen had, die allemaal met alleen Ja of Nee beantwoord konden worden.

Dan bestaan er voor deze enquête 20479 mogelijke patronen van de vorm:

als vraag 1 = Ja en vraag 2 = Nee dan vraag 3 = Ja

Als 10 mensen aan de enquête hebben meegedaan, dan kunnen die op 4140 manieren geclusterd worden. Als we domweg alle mogelijke regels op alle groepen in alle mogelijke clusteringen gaan uitproberen moeten we meer dan 400 miljoen hypothesen testen.

Er zijn twee manieren om deze duizelingwekkende getallen het hoofd te bieden. Ten eerste *domeinkennis*, dat wil zeggen menselijk inzicht, omdat lang niet alle clusteringen en lang niet alle patronen zinvol zijn. Ten tweede *slimme algorithmen* die bijvoorbeeld zien dat als een bepaalde combinatie niet tot een goede oplossing leidt, andere combinaties dat ook niet kunnen doen.

Een data mine tool combineert deze twee mogelijkheden. Het is een interactieve interface naar de database die de gebruiker helpt patronen op te sporen. De gebruiker levert zijn domein-kennis door te vertellen naar welke verbanden hij op zoek is en de interface gebruikt zijn programmatuur om snel zo een verband te vinden.

Om de data mine tool zo efficiënt mogelijk te maken, wordt er bij de ontwikkeling ervan geput uit drie gebieden: databases, statistiek en machine learning—een onderzoeksgebied uit de kunstmatige intelligentie. Databases levert bijvoorbeeld de browsing optimalisatie, het efficiënt omgaan met gerelateerde queries, de statistiek leert welke regels een goede beschrijving vormen en machine learning verteld hoe we snel een goede oplossing kunnen vinden.

Behalve de efficiëntie van de data mine tool, is ook de inzichtelijkheid en presentatie van de gevonden informatie een zwaartepunt van het onderzoek aan het CWI. Immers, de beoogde gebruiker is een manager en geen computer-expert.

2.3 Een Praktijkvoorbeeld van Data Mining

Data mining wordt al in de praktijk toegepast. We sluiten het eerste deel van dit verhaal dan ook af met praktijkvoorbeelden.

Zoals alle disk-drive fabrikanten test IBM haar nieuwe disk-drives uitgebreid voor ze deze naar haar klanten stuurt. De meeste tests zijn eenvoudig en worden tijdens verschillende fasen van de assemblage uitgevoerd. De laatste test is een uitgebreide duurtest als het apparaat gereed is. Deze laatste test kost veruit het meeste tijd en geld. Er zou behoorlijk bespaart kunnen worden als het van te voren te voorspellen zou zijn welke disk-drives waarschijnlijk niet in orde zijn.

Om deze mogelijkheid te onderzoeken hebben medewerkers van IBM een database opgezet waarin zij per disk-drive alle test-resultaten verzamelden. Vervolgens is met behulp van data-mine technieken gezocht naar een verband tussen de resultaten van de eerste goedkope tests en de uiteindelijke dure test voor disk-drives die de laatste test niet haalden.

Hierbij is een regel gevonden die ongeveer 10% van de uiteindelijk falende disk-drives al van te voren herkende. Het gebruik van deze regel bespaart dus aanzienlijk bij het uitvoeren van de laatste, dure, test.

Een ander voorbeeld wordt geleverd door het Britse bankiershuis TSB. Deze heeft regels afgeleid uit gegevens van oude leningen, waarmee de kredietwaardigheid van nieuwe klanten wordt beoordeeld. De nauwkeurigheid van deze regels verbeterde de tot dan toe gebruikte

regels met 3%.

Belangstelling uit het bankwezen en de accountancy moge ook blijken uit de plannen die Barclays Bank en Anderson Consulting hebben voor een data mining onderzoek naar onder meer het analyseren van geldstromen.

3. Hoe werkt Data Mining?

Een data mine tool is dus een systeem dat uit een database nieuwe informatie destilleert in de vorm van karakteristieke patronen. We zullen in dit laatste deel beschrijven hoe het construeren van deze patronen in zijn werk gaat.

We hebben al gezien dat data mining in twee fasen gebeurt: eerst definieert de gebruiker een *klasse* in de gegevens en vervolgens zoekt het systeem voor deze klasse een *karakteristiek patroon*. We bespreken hoe een klasse gedefinieerd wordt, en daarna zullen we uitleggen hoe we het karakteristieke patroon voor de klasse vinden.

Als laatste zullen we ingaan op een aantal problemen, inherent aan het gebruik van databases, en de oplossingen die hiervoor gevonden zijn.

3.1 Klassen

De gebruiker definieert een klasse, waardoor de database uiteen valt in twee groepen: *positieve* voorbeelden van deze klasse, dus objecten in de database die tot deze klasse behoren, en *negatieve* voorbeelden. Bijvoorbeeld, als we geïnteresseerd zijn in kredietwaardigheid van klanten, dan definiëren we een klasse 'kredietwaardig', negatieve voorbeelden zijn dan de personen op de zwarte lijst, positieve voorbeelden zijn alle andere personen in de database.

De gebruiker zal in het algemeen niet alle objecten met de hand classificeren, maar hij of zij zal het systeem een regel geven waarmee het zelf kan bepalen wat de negatieve en de positieve voorbeelden zijn. Deze regel hangt natuurlijk af van de informatie die de gebruiker zoekt. Een aantal voorbeelden zijn:

3.1.1 Condities op attribuut-waarden Als we karakteristieke kenmerken van patiënten met griep zoeken, kunnen we in een medische database een klasse 'griep' definiëren. Positieve voorbeelden van deze klasse zijn patiënten die als diagnose griep hebben, d.w.z. waarvoor het database-attribuut 'diagnose' de waarde 'griep' heeft, en negatieve voorbeelden zijn de patiënten met een willekeurige andere diagnose.

3.1.2 Meerdere databases We kunnen ook zoeken naar verschillen tussen objecten in meerdere databases, bijvoorbeeld naar het verschil tussen klanten uit de database in de vestiging in Emmen en klanten uit de database in Amsterdam. Daartoe wordt de ene klasse gevormd door de klanten uit Amsterdam en de andere klasse door de klanten uit Emmen en wordt gezocht naar karakteristieke patronen voor beide klassen.

3.1.3 Trends in databases Als we willen weten welke globale veranderingen zich over een bepaalde periode in een database hebben voorgedaan, kunnen we een database met dezelfde database van een periode terug vergelijken. De ene klasse wordt dan gevormd door de huidige database en de andere klasse wordt gevormd door de database van een periode daarvoor. Gezocht wordt naar de karakteristieke patronen voor beide perioden.

Indien beide databases dezelfde objecten bevatten (maar op verschillende momenten), kunnen we een klasse definiëren in termen van dit tijds-aspect. Bijvoorbeeld de klasse 'verdubbelde omzet', die alle bedrijven bevat die in deze periode hun omzet hebben verdubbeld.

3.2 Representatie van patronen

Nu de database is opgedeeld in positieve en negatieve voorbeelden van een bepaalde klasse kan het systeem op zoek gaan naar een karakteristiek patroon: een uitspraak die waar is voor elk positief voorbeeld en onwaar voor elk negatief voorbeeld in de database. Dit patroon – de uitvoer van het data mine systeem – moet bovenal begrijpelijk zijn. Data mining is bedoeld om meer inzicht te krijgen in verbanden tussen gegevens in de database, en complexe en ondoorzichtige patronen dragen daar immers niet toe bij.

We zijn dus eigenlijk meer geïnteresseerd in globale beschrijvingen die de realiteit benaderen, dan in complexe beschrijvingen die de realiteit in detail beschrijven. Dit zullen we illustreren met behulp van een voorbeeld uit de natuurkunde. Immers, wat oneerbiedig kunnen we een fysicus zien als een menselijke data miner. Zijn metingen leveren hem een database waarin hij vervolgens op zoek gaat naar een regelmaat. De gevonden regel wordt vervolgens geformuleerd als een fysische wet.

Een, wederom hypothetische, elektronica-gigant is ongerust geworden over de mogelijke verbanden tussen electro-magnetische velden en bepaalde typen kanker. Hij besluit zijn huisfysicus Grofmann uit te laten zoeken of de electro-magnetische velden rond zijn apparatuur gevaarlijk sterk zijn.

Grofmann besluit dat aangezien er alleen maar stroom het apparaat ingaat, het verband tussen de stroomsterkte en de veldsterkte van belang is. Nu staan hem twee wegen tot zijn beschikking. Of hij gaat op zoek naar het precieze verband, hij vindt dan de wetten van Maxwell:

$$
\begin{aligned}
\nabla \cdot \mathbf{D} &= \rho \\
\nabla \cdot \mathbf{B} &= 0 \\
\nabla \times \mathbf{E} &= -\frac{\partial \mathbf{B}}{\partial t} \\
\nabla \times \mathbf{H} &= \mathbf{i} + \frac{\partial \mathbf{D}}{\partial t}
\end{aligned}
$$

Deze wetten beschrijven het verband tussen stroomsterkte en veldsterkte op een zeer gedetailleerd nivo. Hij kan ditzelfde verband ook kwalitatief formuleren als:

als de stroomsterkte toeneemt dan wordt het veld sterker

Het resultaat van de eerste weg is natuurlijk veel preciezer dan het resultaat van de tweede. Maar, de tweede is veel sneller gevonden. Bovendien is het resultaat van de tweede weg goed genoeg om het door zijn baas verlangde antwoord te geven. Dus als Grofmann hart voor zijn baas heeft kiest hij voor de tweede weg.

Dit voorbeeld laat zien dat we niet op zoek zijn naar de meest precieze karakterisering van de klasse, maar veelal genoeg hebben aan een *adequaat* patroon dat de gewenste informatie geeft. Op het CWI zoeken we dan ook representaties die enerzijds krachtig genoeg zijn om interessante verbanden te beschrijven, en anderzijds eenvoudig te interpreteren zijn. Een bijkomend voordeel van een eenvoudige representatie is ook dat het karakteristieke patroon

sneller gevonden wordt.

3.3 Patronen zoeken

Binnen een bepaalde representatie zijn er veelal een onbeperkt aantal patronen te construeren. Deze verzameling van alle construeerbare patronen noemen we de *zoekruimte* en hierin gaan we dus op zoek naar het karakteristieke patroon dat het best bij de klasse past. Om dit patroon te kunnen vinden voeren we een kwaliteits-functie in, die voor elk patroon aangeeft hoe goed het bij de klasse past, dus in welke mate het waar is voor positieve voorbeelden en onwaar voor negatieve voorbeelden en tevens hoe eenvoudig het is. Het doel is nu om zo snel mogelijk het patroon met de beste kwaliteit te vinden.

Het vinden van dit patroon is het klassieke *zoek-probleem* uit de kunstmatige intelligentie. De zoek-ruimte is voor te stellen als een heuvelachtig landschap, waar de hoogte de kwaliteit (van het patroon op die plaats) aanduidt. Het doel is om zo snel mogelijk de hoogste top – het globale maximum te vinden.

Hiertoe definiëren we operaties die een patroon omzetten in een nabijgelegen patroon, en ons dus in staat stellen om kleine stappen te nemen in het landschap. We gebruiken een *zoek-strategie* om in zo min mogelijk stappen het maximum te vinden. Een mogelijke strategie is de hill-climber strategie, waarbij een willekeurig initieel patroon wordt gekozen, en steeds de operatie wordt uitgevoerd die de grootste kwaliteits-winst oplevert. Het zoeken gaat door totdat geen van de operaties in winst resulteert, en dus een maximum is bereikt.

Een nadeel is echter dat het zoek-proces kan blijven steken in een lokaal maximum – dat het heuveltje is beklommen in plaats van de grote berg. Een oplossing hiervoor is *simulated annealing*, waarbij operaties met een zekere willekeur worden gekozen, zodat het zoekproces kan ontsnappen aan lokale maxima. Gedurende het zoekproces neemt de willekeur af, zodat het zich uiteindelijk in het globale maximum kan stabiliseren.

Alternatieve zoek-strategieën zijn *genetische algoritmen*, gebaseerd op evolutionaire processen. Hierbij wordt niet één initieel patroon gekozen, maar een hele verzameling—ook wel populatie genaamd. Uit deze populatie worden de patronen met de hoogste kwaliteit met elkaar gecombineerd om de volgende generatie te vormen. De zoekruimte wordt zodoende parallel afgezocht, waarbij informatie over verschillende punten in de zoekruimte wordt gecombineerd om nieuwe patronen te genereren.

Beide zoek-strategiën zijn ongeïnformeerd, d.w.z. ze gebruiken geen additionele informatie over de zoekruimte, maar laten de keuze van de operaties alleen afhangen van de kwaliteit van het resulterende patroon. Het zoekproces kan versneld worden door het gebruik van heuristieken, zoals reeds bekende verbanden, randvoorwaarden aan de patronen en gebruikers-interactie.

3.4 Mogelijkheden en onmogelijkheden van Data Mining

Met data mining kan dus op efficiënte wijze interessante informatie uit databases gewonnen worden. Er is echter een complicatie: gegevens in een database bevatten fouten en zijn onvolledig. Om onder deze omstandigheden toch nog succesvol patronen te kunnen vinden moet het data mine proces worden aangepast. We zullen deze problemen nader bespreken, en aangeven welke aanpassingen noodzakelijk zijn.

3.4.1 Ruis Verminkte informatie, oftewel ruis, leidt tot problemen bij de constructie van patronen. Stel dat we een karakteristiek patroon voor een klasse met een aantal verminkte voorbeelden proberen te creëeren. Omdat dit patroon karakteristiek is voor deze verzameling, zal het ook voor de verminkte voorbeelden gelden. Dit is natuurlijk niet de bedoeling, en daarom moeten verminkte voorbeelden uit de klasse verwijderd worden. Helaas is het in het algemeen niet mogelijk om verminkte voorbeelden van niet verminkte te onderscheiden. We maken daarom gebruik van statistische technieken, zoals de vuistregel dat een kleine hoeveelheid voorbeelden, die sterk afwijken van de rest van de voorbeelden, veroorzaakt worden door ruis en dus genegeerd moeten worden.

Bovendien zijn patronen globale beschrijvingen van data, waarin een zekere redundantie verwerkt is. Hierdoor is een speling in de gegevens mogelijk, het systeem is *fout-tolerant.* Uit tests blijkt dat zelfs databases die grotere hoeveelheden ruis bevatten door het systeem redelijk accuraat behandeld worden. Dit wordt ook wel *graceful degradation* genoemd, de performance van het systeem gaat slechts geleidelijk achteruit als de hoeveelheid ruis toeneemt.

3.4.2 Onvoldoende Informatie Om precieze regels af te leiden, moet alle eventueel belangrijke informatie in de database voor handen zijn. Voor sommige toepassingen is dit geen realistische aanname, bijvoorbeeld of iemand schade zal rijden is van veel meer persoonskenmerken en factoren afhankelijk, dan die, die zijn opgeslagen in de database. Evenzo is bijvoorbeeld een medische diagnose nooit met volledige zekerheid af te leiden uit persoons- en klinische-gegevens in een database.

Dit probleem van onvoldoende informatie is in principe onoverkomelijk. Er kan meer informatie in de database gestopt worden, maar omdat het aantal relevante factoren te groot en veelal ook onbekend of onmeetbaar is, kunnen de regels nooit deterministisch zijn. Een oplossing is dan ook niet-deterministische regels te construeren, zogenaamde *waarschijnlijkheidsregels*, die een uitspraak doen over de kans dat een bepaald object tot een klasse behoort, indien het aan bepaalde voorwaarden voldoet.

Bijvoorbeeld kan uit een database een regel worden afgeleid, die stelt dat indien een persoon aan bepaalde voorwaarden voldoet (het karakteristieke patroon), dat deze persoon dan met n procent zekerheid tot de risico groep behoort. Deze n procent, de waarschijnlijkheid van de regel, kan worden berekend door voor de database te controleren in hoeveel procent van de gevallen de regel correct classificeert.

4. Conclusie

Samenvattend kunnen we zeggen dat data mining een middel is waarmee nuttige informatie uit een database gehaald kan worden, informatie die anders verborgen blijft. De technieken die hiervoor gebruikt worden zijn al langer bekend, maar worden pas recentelijk voor grote verzamelingen gegevens – zoals databases – gebruikt. De belangrijkste redenen hiervoor zijn dat enerzijds databases zo omvangrijk worden dat het overzicht verloren gaat en geavanceerde data mine technieken nodig zijn om dit overzicht weer terug te krijgen. Anderzijds zijn computers nu zo krachtig dat ook in grotere verzamelingen succesvol ge'mined' kan worden. Daarom staat data mining niet alleen bij het CWI, maar ook bij andere vooraanstaande onderzoeksinstituten volop in de belangstelling.